

# Procesamiento de imágenes solares para la obtención de mapas sinópticos

---

## Processing solar images for obtaining synoptic maps

Gabriel García García



Trabajo de Fin de Grado en Ingeniería Informática  
Facultad de Informática, Universidad Complutense de Madrid

Madrid, junio de 2020

Directores:

Bernabé García, Sergio  
Vaquero Martínez, José Manuel



# Agradecimientos

Gracias a mi familia por apoyarme, inspirarme, enseñarme y aguantar todo de mí. A mi gran equipo de compañeros, quienes hemos logrado sobrevivir a todas las diversas experiencias que la carrera nos ha aportado durante todos estos años.

A su vez, quiero agradecerles a mis directores. A Sergio, quien se ha comportado como profesor, director y compañero a la hora de abordar este proyecto, y a José Manuel, por brindarnos de todo el conocimiento necesario para poder hacerlo realidad.





# Índice general

Índice general	I
Índice de figuras	IV
Índice de tablas	VII
Resumen	VIII
Abstract	IX
<b>1. Introducción</b>	<b>1</b>
1.1. Motivación . . . . .	1
1.2. Objetivos . . . . .	3
1.3. Plan de trabajo . . . . .	4
1.4. Organización de esta memoria . . . . .	6
<b>2. Análisis de imágenes solares</b>	<b>9</b>
2.1. Imagen astronómica . . . . .	9
2.2. Sensor de imagen . . . . .	11
2.3. Objetivos de una imagen pasabanda . . . . .	14
2.3.1. Adquisición monobanda . . . . .	14
2.3.2. Adquisición multibanda . . . . .	15
<b>3. Sistemas de coordenadas y orientación de imágenes</b>	<b>19</b>
3.1. Sistemas de coordenadas geocéntricos . . . . .	20
3.1.1. Sistemas de coordenadas geocéntrico terrestre . . . . .	20
3.1.2. Sistemas de coordenadas heliográficas . . . . .	21

3.2.	Proyecciones de sistemas de coordenadas . . . . .	25
3.2.1.	Proyección cenital de un sistema tridimensional heliográfico . . . . .	26
<b>4.</b>	<b>Mapas sinópticos</b>	<b>31</b>
4.1.	Proyección a coordenada cilíndrica . . . . .	32
4.1.1.	Fernández Guasti Squire . . . . .	33
4.1.2.	Simple Stretching . . . . .	34
4.1.3.	Elliptical Grid Mapping . . . . .	36
4.2.	Fragmentación de imágenes . . . . .	38
4.3.	Composición y validación . . . . .	41
<b>5.</b>	<b>Desarrollo de la aplicación</b>	<b>45</b>
5.1.	Planificación del desarrollo . . . . .	45
5.1.1.	Lenguaje de programación . . . . .	45
5.1.2.	Entorno de desarrollo . . . . .	47
5.1.3.	Repositorio . . . . .	50
5.2.	Gestión del proyecto . . . . .	50
5.2.1.	Fase de investigación sobre paquetes . . . . .	51
5.2.2.	Estructuración del proyecto . . . . .	52
5.2.3.	Sistema de la información del modelo . . . . .	55
<b>6.</b>	<b>Funcionalidad de la aplicación</b>	<b>60</b>
6.1.	Agrupamiento de eventos . . . . .	60
6.2.	Funcionalidades principales . . . . .	61
6.2.1.	Servidores de consulta de los instrumentos . . . . .	61
6.2.2.	Cargar una imagen o secuencia a la aplicación . . . . .	63
6.2.3.	Aplicar un filtro sobre un conjunto de imágenes . . . . .	64
6.2.4.	Creación de un mapa sinóptico . . . . .	65
6.3.	Funcionalidades secundarias . . . . .	67

6.3.1. Toppings . . . . .	67
6.3.2. Histogramas . . . . .	69
6.3.3. Diagrama DFT . . . . .	71
6.3.4. Opciones de vista sobre la imagen . . . . .	72
6.3.5. Secuencia de imágenes . . . . .	73
<b>7. Conclusiones y trabajo futuro</b>	<b>75</b>
7.1. Conclusiones . . . . .	75
7.2. Trabajo futuro . . . . .	76
<b>Bibliografía</b>	<b>78</b>
<b>A. Introduction</b>	<b>79</b>
A.1. Motivation . . . . .	79
A.2. Objectives . . . . .	81
A.3. Work plan . . . . .	82
A.4. Organization of this memory . . . . .	82
<b>B. Conclusions and future work</b>	<b>86</b>
B.1. Conclusions . . . . .	86
B.2. Lines of future work . . . . .	87
<b>C. Estudios reales haciendo uso de la aplicación</b>	<b>89</b>
C.1. Mejora de nitidez sobre imagen SDO/AIA . . . . .	89
C.2. Perfilado de estructuras . . . . .	94
C.3. Procesamiento de un mapa sinóptico . . . . .	95

# Índice de figuras

1.1. Plan de trabajo. . . . .	5
2.1. Esquema formato FITS. . . . .	10
2.2. Proceso de digitalización por parte de un sensor cromático. Fuente: TilanoTV. . . . .	12
2.3. Colormaps disponibles para imágenes de distintos instrumentos solares. . . . .	13
2.4. Imágenes brindadas en distintos espectros por el instrumento AIA de la sonda SDO. . . . .	15
2.5. Procesamiento de las imágenes de la Nebulosa de la Laguna obtenido con la sonda Hubble. . . . .	16
2.6. Magnetograma del Sol tomado por el instrumento HMI de la sonda SDO el 25 de enero de 2017. . . . .	17
3.1. Representación de un sistema de coordenadas geocéntricas terrestres. . . . .	21
3.2. Representación de la velocidad angular en función de la latitud en valor absoluto. . . . .	22
3.3. Representación de los movimientos de precesión y nutación. . . . .	23
3.4. Representación de un sistema heliocéntrico. . . . .	24
3.5. Proyección de un sistema de coordenadas heliográficas sobre un plano. . . . .	26
3.6. Variación de los ángulos de inclinación del Sol. . . . .	27
3.7. Coordenadas polares. . . . .	28
3.8. Método de posicionamiento mediante uso de plantillas. . . . .	30
4.1. Representación de rotaciones solares mediante el diagrama de Thilo. . . . .	32
4.2. Representación de la superficie terrestre mediante dos proyecciones cenitales. . . . .	33
4.3. Proceso FGS. . . . .	33

4.4. Método FGS sobre meridianos. . . . .	34
4.5. Proceso simple stretching. . . . .	35
4.6. Proceso simple stretching sobre meridianos. . . . .	36
4.7. Proceso elliptical squircle. . . . .	36
4.8. Proceso elliptical squircle sobre meridianos. . . . .	37
4.9. Aplicación del método FGS a una imagen solar real. . . . .	37
4.10. Simulación de la rotación diferencial para 4 días. . . . .	38
4.11. Composición de un mapa sinóptico a través de una secuencia de imágenes comprendidas entre el 1 al 28 de enero de 2017. . . . .	41
4.12. Mapa sinóptico correspondiente a la rotación de Carrington 2114 realizado por A.Hamada. . . . .	42
4.13. Mapa sinóptico correspondiente al realizado por la aplicación. . . . .	43
5.1. Sistema de entornos de desarrollo en Anaconda. . . . .	48
5.2. Sistema de paquetes en Anaconda. . . . .	49
5.3. Interfaz gráfica de PyCharm. . . . .	50
5.4. Diagrama patrón MVC. . . . .	54
5.5. Paso de información entre la vista y el controlador. . . . .	55
5.6. Paso de información entre el parserEvents y el controlador. . . . .	56
5.7. Solicitud del objeto SunPy al sistema contenedor. . . . .	57
5.8. Ejecución de un comando por parte del gestor del modelo. . . . .	57
5.9. Declaración de una estructura como observador. . . . .	58
5.10. Actualización de los observadores tras una modificación sobre el sistema con- tenedor. . . . .	58
6.1. Cargar imágenes en un objeto FITS. . . . .	63
6.2. Imagen procedente del instrumento SDO/AIA 171 Ångstrom del día 17/08/2017 cargada en la interfaz. . . . .	65

6.3. Aplicación de un ajuste gamma. . . . .	66
6.4. Composición de un mapa sinóptico a través de una secuencia de imágenes. . .	66
6.5. Aplicación de una malla sobre la imagen. . . . .	67
6.6. Cromosfera en la imagen. . . . .	68
6.7. Limbo sobre la imagen. . . . .	69
6.8. Contornos sobre imágenes solares. . . . .	70
6.9. Histograma de una imagen solar. . . . .	70
6.10. Espectro de frecuencias de una imagen solar. . . . .	71
6.11. Aplicación de diferentes pseudocolores sobre la imagen. . . . .	72
6.12. Representación de una secuencia de imágenes. . . . .	73
A.1. Work plan. . . . .	83
C.1. Menú de carga. . . . .	90
C.2. Elección de fichero FITS. . . . .	91
C.3. Muestra en el frame el fichero FITS escogido. . . . .	91
C.4. Imagen base sobre la que se realizará el procesado de mejora de nitidez. . . .	92
C.5. Mejora de brillo selectivo sobre la imagen base. . . . .	92
C.6. Diferentes secciones del resultado tras el procesado de nitidez. . . . .	93
C.7. Aplicación de una ecualización local de parámetro 15. . . . .	94
C.8. Resultado final del perfilado de las distintas estructuras encontradas en la imagen tratada. . . . .	95
C.9. Imagen procedente del instrumento SDO/AIA 4500 Ångstrom tomada el día 06/01/2015. . . . .	96
C.10. Aplicación de un ajuste gamma para reducir el exceso de luminosidad. . . . .	96
C.11. Aplicación de un ajuste gamma para reducir el exceso de luminosidad. . . . .	97

# Índice de tablas

6.1. Consulta sobre el cliente que contiene los ficheros de la sonda SDO. . . . .	62
---	----

# Resumen

Con los avances de la ingeniería, se han logrado crear una gran serie instrumentos de gran calidad y precisión, orientados a la observación y a la obtención de parámetros sobre cuerpos celestes. Debido a que el Sol es la estrella más cercana, hoy en día existen numerosas investigaciones y ramas de estudio procesando grandes volúmenes de datos a tiempo real de este cuerpo celeste, procedentes de observatorios y sondas espaciales.

El registro de movimientos, distancias y estructuras entre otros sobre el Sol crea la necesidad de tener a disposición un sistema de posicionamiento preciso. A lo largo del proyecto se mostrarán los distintos sistemas de posicionamiento aplicables al Sol y sus métodos de proyección que facilite su aplicabilidad directa sobre las distintas imágenes.

En otras ocasiones se necesitará comparar estructuras y patrones sobre varios periodos de actividad solar. Para lograr este objetivo, se almacenarán distintas rotaciones en un único mapa haciendo uso de una secuencia de imágenes permitiendo, a su vez, el posicionamiento de cada elemento observable sobre el mismo. Estos mapas se denominan mapas sinópticos y es el objetivo principal del proyecto, pudiéndose aplicar haciendo uso de la aplicación creada con dicha finalidad.

## Palabras clave

Sol, física estelar, sonda espacial, observatorio, mapa sinóptico, proyecciones, sistemas de posicionamiento.



# Abstract

With the advances of the engineering, they have succeeded to create a great series of instruments of high quality and precision, oriented to the observation and the obtaining of parameters on celestial bodies. Because the Sun is the nearest star, today there are numerous investigations and branches of study processing large volumes of real-time data from this celestial body, from observatories and space probes.

The recording of movements, distances and structures among others creates the need to have a precise positioning system available. Throughout the project, the different positioning systems applicable to the Sun and their projection methods will be shown to facilitate their direct applicability to the different images.

At other times you will need to compare structures and patterns over several periods of solar activity. To achieve this objective, different rotations will be stored on a single map using a sequence of images, allowing, in turn, the positioning of each element observable on it. These maps are called synoptic maps and are the main objective of the project and can be applied using the application created for this purpose.

## Keywords

Sun, stellar physics, space probe, observatory, synoptic map, projections, positioning systems.

# Capítulo 1

## Introducción

### 1.1. Motivación

Durante la historia de la astronomía [1, 2], se tiene constancia que todo registro de datos celestes era almacenado en papel o piedra. La única forma de seguimiento y estudio de objetos celestes hasta el siglo XVII era mediante el método observacional, un hecho de ello son las civilizaciones orientales, las cuales almacenaban grabados sobre sus observaciones sobre el Sol. Estas civilizaciones usaban usualmente el Sol para la regulación de su calendario. Para ello, lo observaban cuando la posición del Sol era próxima al horizonte siempre que se cumplieran con unas condiciones meteorológicas apropiadas (como niebla o tormentas de polvo entre otras) haciendo que la atmósfera actuara de filtro solar.

Con el avance de la óptica a finales del siglo XVI y comienzos del XVII, surgieron ciertos instrumentos de observación para distancias lejanas, aunque en su mayoría eran pensadas para su uso en guerra. La invención del telescopio es algo difusa, inicialmente fue atribuida al alemán Hans Lippershey, aunque estudios más recientes del siglo XX sugieren que fue el catalán Joan Roget. En 1609, tras exponer Galileo Galilei el primer telescopio de refracción orientado a la observación de los astros a la comunidad científica del entonces, surge una nueva época de descubrimientos en la historia astronómica apareciendo nuevos campos de estudio e investigación aplicados al Sol.

A lo largo del siglo XVII empezaron a surgir distintos tipos de telescopios, todos ellos

seguían unos patrones similares, aunque la disposición de sus lentes variaba. Tras ello, en 1611 comenzó la carrera de las manchas solares. El alemán Christopher Scheiner inventó diversas técnicas para la observación del Sol enfocándose en métodos que permitieran visualizarlo de una manera segura (mediante filtros o proyecciones sobre una membrana, como si de una diapositiva se tratase).

Scheiner supuso que dichas manchas debían ser proyecciones de astros sobre su superficie o defectos en el propio ojo o de la lente al observar. En el entonces se sugería que todo cuerpo celeste era perfecto. Galileo en cambio, lo describió como protuberancias en la superficie del Sol, de gran tamaño, que poseían movimiento. Este pensamiento le llevó a lo correcto, permitiéndole calcular los primeros datos astronómicos del Sol tales sean su periodo y su velocidad de rotación entre otros [3].

Estudios más completos y precisos sobre las manchas solares no comenzarán hasta casi 2 siglos hacia delante junto al alemán Heinrich Schwabe. Schwabe realizó registros diarios de manchas solares desde el año 1826 hasta 1843. Durante este lapso, descubrió una periodicidad en el número de manchas cada 11 años a lo que denominó ciclo. Este hecho dio lugar a la necesidad de cuantificar la actividad solar, no solo del siglo XIX, sino de años anteriores haciendo uso de notas y cuadernos de hasta alrededor del 1745 para poder cuantificar y explicar con cierta precisión su hallazgo.

El suizo Johann R. Wolf intrigado, comenzó en 1847 un conteo de manchas y poros, al que le asignaba en su conjunto un valor fácilmente calculable, conocido como número de Wolf. Aunque existen variaciones como el conteo por grupos (1998), ambos son usualmente aplicados a día de hoy. El número de Wolf permite abstraer en un número las observaciones de cada día, registrando únicamente un valor por observación.

El 1 de septiembre de 1859 es registrada la posible tormenta solar más grande de la historia por Richard C. Carrington (este acontecimiento es conocido como evento Carrington). Al realizar la observación se descubre por primera vez una gran explosión junto a la salida de una gran llamarada proveniente del Sol, conocidas como CME (Coronal Mass Ejection).

Tras la observación de dicho evento, al día siguiente fue detectada una tormenta geomagnética sobre la Tierra acompañada de auroras a latitudes bajas, indicando a los científicos del siglo XIX que las eyecciones solares aparentemente podría estar relacionadas con estos fenómenos. Este fenómeno será estudiado posteriormente en 1931 por Sydney Chapman y Vincenzo Ferraro. Carrington realizó numerosas investigaciones con una exactitud increíble, lo que permite que sus técnicas se puedan seguir usando a día de hoy. De entre ellas se encuentran: el sistema de coordenadas heliográficas y la rotación diferencial en función de la latitud del cuerpo. Estos dos temas serán abordados a lo largo de la memoria.

Uno de los mayores avances en el campo de la astronomía solar fue realizado por el astrónomo estadounidense George Ellery Hale, quién en 1892 idea el espectroheliógrafo, un instrumento que permite observar al Sol en una banda del espectro electromagnético. Hale continuó en paralelo con el estudio de las tormentas geomagnéticas y llegó a la conclusión mediante lecturas de heliomagnetogramas en 1903 que las manchas solares están asociados a campos magnéticos. Desde entonces, han surgido una gran variedad de nuevos estudios derivando en investigaciones como el viento solar, radiación cósmica, auroras, emisiones de rayos X, etcétera.

Desde el descubrimiento de los ciclos solares hasta la actividad magnética del Sol entre otros muchos estudios variados, para lograr comparar resultados de una forma más óptima en todos ellos, ya sea mediante dibujos o lecturas de magnetogramas, se registra en un único mapa o esquema varias rotaciones del sol permitiendo comparar sus elementos observables en distintos lapsos de tiempo. Estas representaciones son comúnmente conocidas como planisferios solares o mapas sinópticos del Sol y es la finalidad de este proyecto.

## 1.2. Objetivos

El objetivo de este trabajo es el desarrollo de una aplicación que permita manejar imágenes solares y desarrollar mapas sinópticos. Esta aplicación contará con diversas herramientas que permitan de una forma sencilla y potente llegar al cometido descrito.

Para lograr el objetivo final se abordarán en la memoria los siguientes objetivos específicos, los cuales se enumeran a continuación:

- Manipulación de ficheros *FITS* (Flexible Image Transport System).
- Análisis y comprensión de los diferentes sistemas de posicionamiento sobre el Sol.
- Aplicación de distintos sistemas de coordenadas para situar una estructura sobre una imagen.
- Elaboración de un sistema capaz de crear mapas sinópticos a partir de una secuencia de imágenes.
- Validación de mapas sinópticos obtenidos a partir de imágenes similares por expertos.
- Desarrollo de una aplicación capaz de componer mapas sinópticos y de disponer de una serie de herramientas sencillas de edición.

### 1.3. Plan de trabajo

Se estructurará el trabajo por distintas actividades principales dictadas por los objetivos a cumplir, los cuales se irán complementando a lo largo del desarrollo del proyecto. Dichas actividades se dividirán en cuatro grupos principales, de entre ellas:

- *Fase inicial*: se describirá la estructura general del proyecto.
- *Fase de investigación*: se obtendrá la información necesaria y se pondrá en práctica para el debido desarrollo de la memoria y el proyecto de programación.
- *Fase de desarrollo de la aplicación*: se desarrollarán las distintas componentes de la aplicación.
- *Fase de desarrollo de la memoria*: se redactarán las distintas partes que compondrán la memoria.

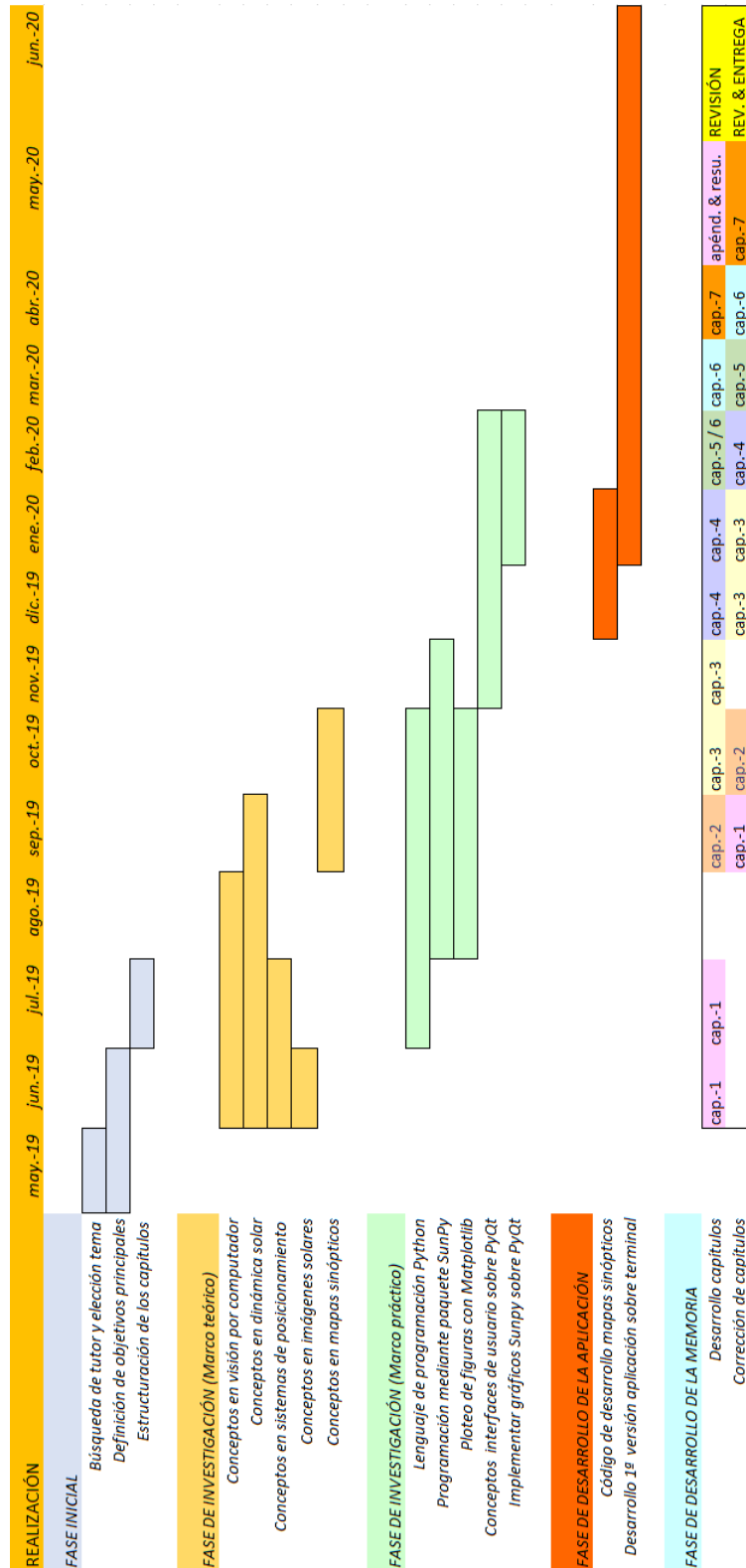


Figura 1.1: *Plan de trabajo.*

## 1.4. Organización de esta memoria

La memoria se estructurará en los siguientes capítulos principales, recogiendo cada uno las siguientes competencias:

- **Introducción:** se realizará un análisis breve histórico de la física solar, se mostrarán los objetivos del proyecto y se analizarán las distintas competencias brindadas por cada capítulo.
- **Análisis de imágenes solares:** será en el segundo capítulo donde se tratarán las imágenes astronómicas y el formato *FITS* utilizado para su almacenamiento. Se explicarán los sensores usados para obtener las imágenes.
- **Sistemas de coordenadas y orientación de imágenes:** en el capítulo tercero se hará mención de los sistemas de coordenadas geocéntricas y heliográficas, los movimientos que afectan al Sol y las distintas formas de proyectar los distintos sistemas de coordenadas. A su vez se mostrará cómo posicionar un punto sobre una imagen solar y cómo orientarla debidamente.
- **Mapas sinópticos:** en el capítulo 4 se hará alusión a los sistemas de proyección en coordenadas cilíndricas y al uso de sistemas basados en *squircle* para realizar dichas proyecciones sobre imágenes solares. Se enseñará el método de composición y la validación de estos mapas.
- **Desarrollo de la aplicación:** se plantearán los elementos necesarios para la propia estructura de la aplicación a lo largo del capítulo 5.
- **Funcionalidad de la aplicación:** se mostrarán las funcionalidades principales y herramientas de las que dispone la aplicación. Todo ello será descrito en el capítulo 6.
- **Conclusiones y trabajo futuro:** estarán descritas en el último capítulo. Se mostrarán las distintas conclusiones y hechos realizados a lo largo del Trabajo Fin de Grado

y sus posibles implementaciones para trabajos futuros.





# Capítulo 2

## Análisis de imágenes solares

### 2.1. Imagen astronómica

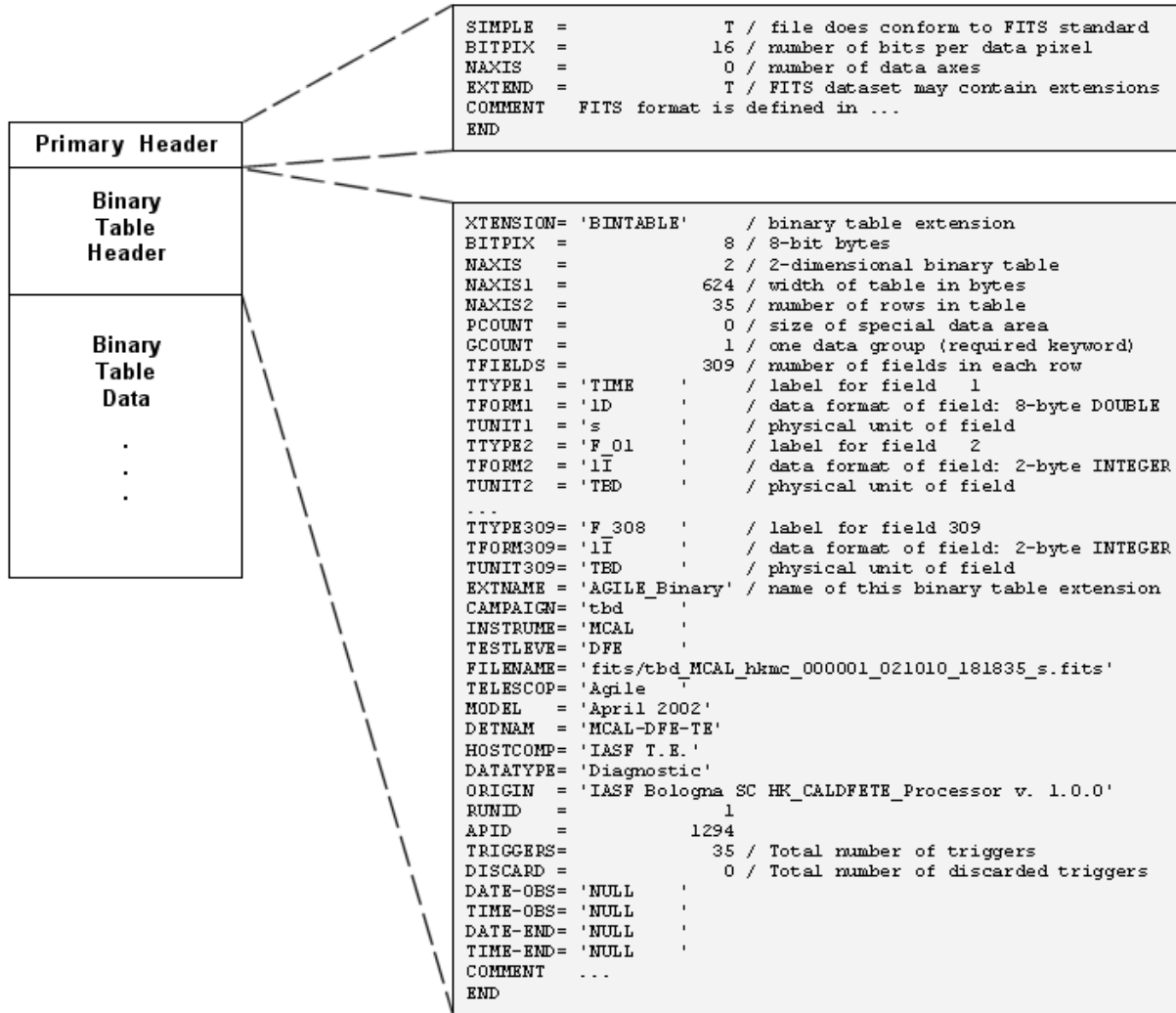
Desde el siglo XVII, la forma más habitual de observación astronómica fue mediante telescopios, haciendo uso de distintos filtros y/o proyecciones sobre una pantalla donde se marcaban objetivos de la observación. Debido a la evolución de los sensores, las técnicas de ingeniería y la informática, la astronomía observacional se ha digitalizado en su mayor rango haciendo uso de imágenes multiespectrales.

Una imagen digital es una representación multidimensional de un sistema, haciendo uso de la radiación que emite en el espectro electromagnético.

Hasta 1980, toda la información generada por los instrumentos era almacenada en un archivo local con el formato propio del observatorio. Estos archivos ya procesados en su mayoría eran mandados a otras instituciones y observadores para un análisis secundario. Esto suponía la necesidad de tener una herramienta de conversión de formato debido a la falta de un estándar. A finales de la década de 1970, se establece el formato *FITS* (Flexible Image Transport System), un formato estándar aún mantenido por la NASA y la Unión Astronómica Internacional.

El formato *FITS* se establece como un sistema creado por bloques, cada uno compuesto por una cabecera (que hace referencia a los datos almacenados en su mismo bloque) y los datos almacenados. Cada cabecera se compone de tarjetas o cards (un sistema de clave-

valor que pueden contener a su vez comentarios) y la parte de almacenamiento principal del bloque, que contiene desde imágenes bidimensionales hasta hipercubos y series entre otros tipos. Esto permite un gran escalado horizontal a la hora de almacenar información, debido a que podemos almacenar en un único archivo información de distintos instrumentos referentes a un mismo sistema de observación.



**Figura 2.1:** Esquema formato FITS.

Tal y como puede observarse en la Figura 2.1, el formato empieza por el *Primary Header*. Esta cabecera es indispensable debido a que es donde se indica qué estándar FITS se usa, y posee los campos clave-valor necesarios para que pueda ser leído el fichero por el

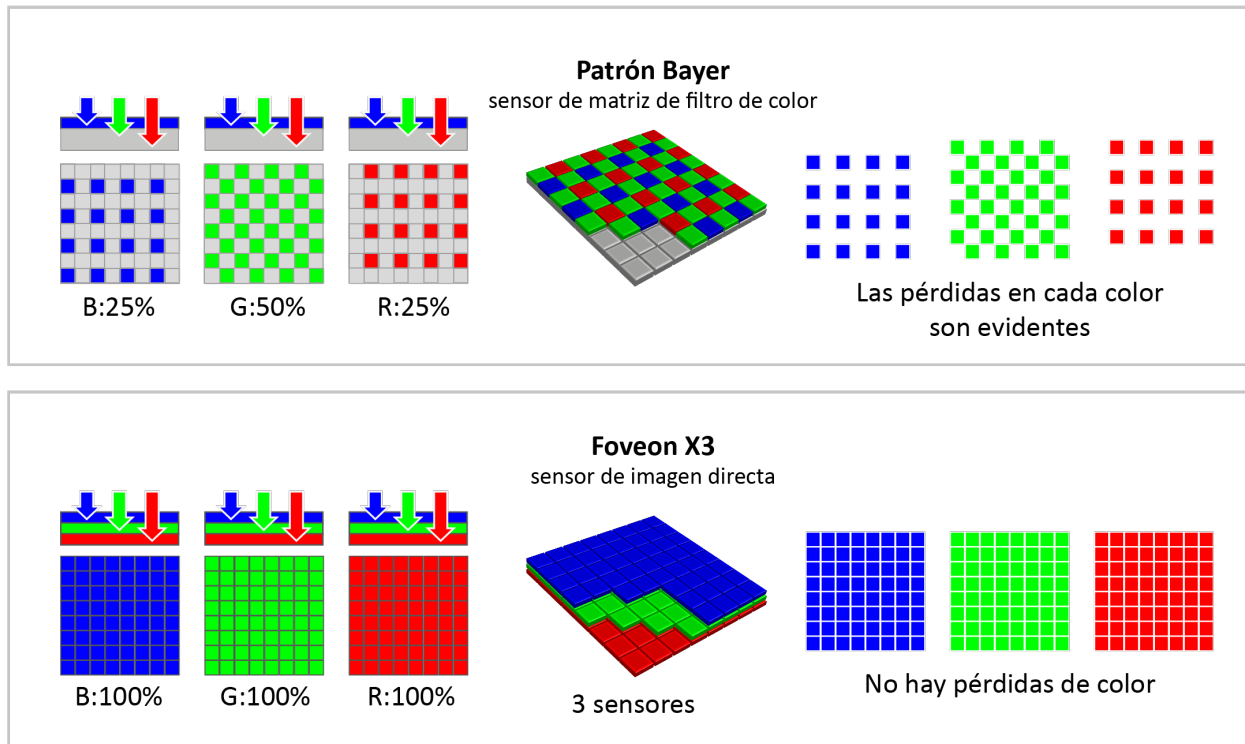
programa. Lo siguiente es la *Binary Table Header*. Esta cabecera, la cual contiene en este caso información sobre los datos que almacena el archivo, puede contener datos históricos, temporales, de movimiento, etcétera. Una vez se declara una cabecera, se almacenan sus datos correspondientes. Dado este proceso existe un bloque de datos por cada cabecera.

## 2.2. Sensor de imagen

El sensor de imagen acoplado al instrumento será el encargado de obtener las imágenes. Estos sensores son en su mayoría de clase *CCD* (*Charge-Coupled Device*) o *CMOS* (*Complementary Metal-Oxide-Semiconductor*) y pueden ser cromáticos o monocromáticos [4]. Estos sensores están compuestos por elementos semiconductores y fotosensibles, en forma de malla y su funcionamiento se basa en el efecto fotoeléctrico. La funcionalidad principal de un sensor fotoeléctrico es transformar un cambio de intensidad de radiación electromagnética a una señal analógica que pueda ser tratada posteriormente.

Se denomina espectro electromagnético a todo el rango energético al que puede pertenecer una onda electromagnética, catalogada en función de su frecuencia (bajo las unidades de Hz) o su longitud de onda (en unidades de metro o Ångstrom [Å] para medidas más pequeñas). El ser humano posee un sistema de visión capaz de distinguir colores en el espectro visible (entre los 350 a los 750 nm), es por ello que existen una gran variedad de sistemas cuya finalidad es recrear este rango de colores. Ejemplo de ello son: el sistema *HSV* (*Hue, Saturation and Value*), *HSL* (*Hue, Saturation and Lightness*) o *RGB* (*Red, Green and Blue*). El sistema RGB es el más usado y se compone de tres arrays bidimensionales los cuales cada uno de ellos representa un rango de valores en las longitudes de onda del rojo, verde y azul.

Los sensores cromáticos hacen uso de este sistema mediante los mosaicos de Bayer (ver Figura 2.2). A cada fotorreceptor se le superpone un filtro que solo deja pasar ciertas longitudes dentro de un rango del espectro visible, asignando un voltaje a cada valor del rango [5, 6]. El ser humano es más sensible a ciertos rangos del espectro de tonalidades verdes, que a los rojos o los azules. Es por ello que en un sensor cromático, el 50 % de los píxe-



**Figura 2.2:** *Proceso de digitalización por parte de un sensor cromático. Fuente: TilanoTV.*

les sean destinados para tonalidades verdes, el 25 % para rojos y el otro 25 % para azules siempre que usen el sistema RGB. Esto limita el área del sensor reduciendo la resolución de la imagen, debido a que existe un proceso de debayerización donde los huecos se interpolan por cercanía de celdas provocando que existan alternativas a este proceso. Los sensores cromáticos basados en multicapas de absorción (ver Figura 2.2 inferior), apilan tres arrays de fotorreceptores aprovechando la capacidad de traspaso del silicio, para crear una imagen bidimensional por cada capa. De esta forma, no se perderá profundidad ni resolución en comparación a los sensores basados en mosaicos de Bayer.

En el campo de la astronomía, el estudio espectral de los cuerpos celestes no se concreta en el espectro visible sino que hace objeto de estudio otros espectros como el ultravioleta o el infrarrojo. Los sensores aplicados para ello no podrán seguir una estructura cromática, debido a que no puede mostrarse su color en un espectro que no podemos observar. Los sensores monocromáticos, en función de su sensibilidad y su material de fabricación, nos



## 2.3. Objetivos de una imagen pasabanda

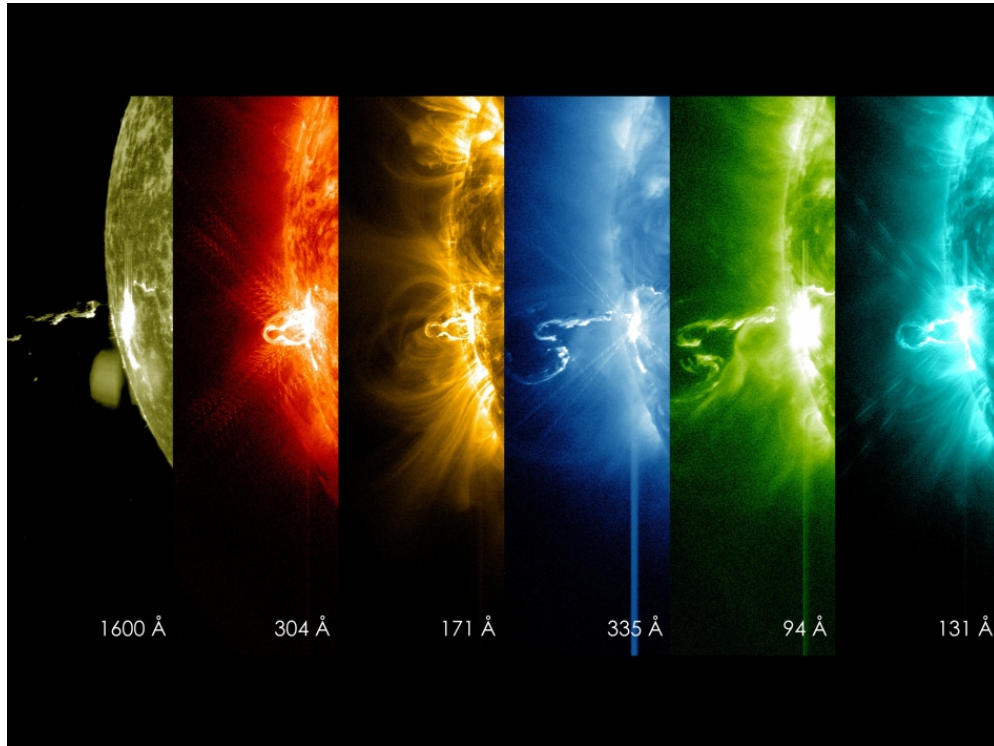
La astrofotografía agrupa las imágenes generalmente en dos conjuntos: el primero para obtención de información aplicada en términos de investigación (aquellas provenientes de observatorios) y en el segundo, las detallistas para exposición (de carácter general) [7].

Generalmente las imágenes provenientes de una sonda espacial o un observatorio se somete a un conjunto de operaciones de procesamiento sobre correcciones de luminosidad, posición y realce entre muchos otros métodos.

### 2.3.1. Adquisición monobanda

Las técnicas monocromáticas se concretan sobre un elemento o una franja del espectro electromagnético, permitiendo un estudio monobanda. Estos estudios son comunes en el Sol, en los que existen elementos en una mayor concentración que otros. Cuando filtramos por ejemplo la luz del Sol dejando pasar la radiación electromagnética asociada a la banda del Hidrógeno mediante un filtro  $H\alpha$ , este nos permitirá observar ciertas partes de la estructura del Sol como la cromosfera y sus elementos representativos (una capa de muy poca densidad y que es difícil de detectar). A su vez, si usamos bandas de elementos muy ionizados como por ejemplo el Hierro X, pueden observarse capas de mucha intensidad como la corona solar (además de concretar temperaturas mediante las propiedades de la banda) entre otras.

Existen numerosas sondas enviadas para el estudio del Sol tales como la misión *SOHO* (*Solar and Heliospheric Observatory*), *SDO* (*Solar Dynamics Observatory*) y la *Solar Parker*. De entre ellas haremos uso de las imágenes brindadas por la sonda *SDO* mediante sus instrumentos *AIA* (*Atmospheric Imaging Assembly*), el cual nos brindará de imágenes en distintas bandas del espectro tal y como puede observarse en la Figura 2.4, y el instrumento *HMI* (*Helioseismic and Magnetic Imager*), el cual brinda imágenes con información de campos magnéticos del Sol como se puede apreciar en la Figura 2.6.



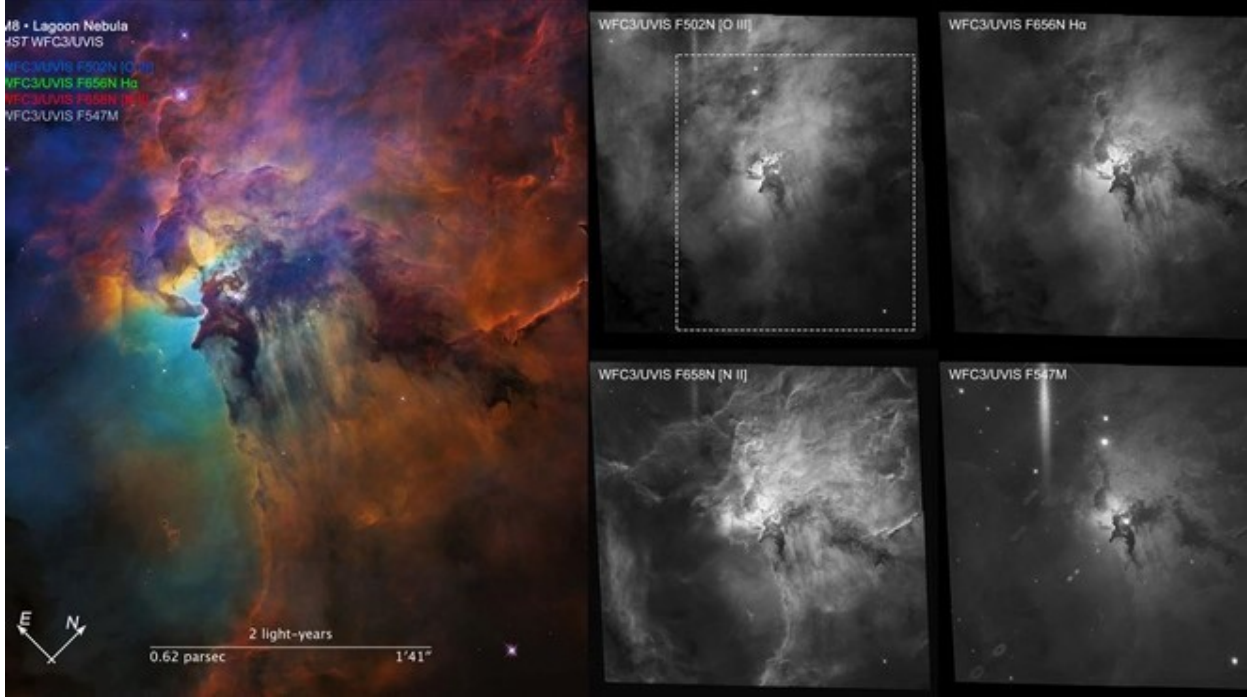
**Figura 2.4:** *Imágenes brindadas en distintos espectros por el instrumento AIA de la sonda SDO.*

### 2.3.2. Adquisición multibanda

Las imágenes multibanda se componen de una serie de capas o arrays los cuales variarán en función del estudio que se quiera aplicar. La selección de cada capa varía en gran medida del equipo y de la finalidad del estudio. Es por eso que cuando las imágenes son de carácter general, se clona una imagen en tres capas y se modifican por separado para obtener un canal de color por cada una (siempre que se use un sistema de tres canales como el RGB descrito anteriormente). Este proceso no brinda un proceso informativo, debido a que cada imagen por canal es modificada a intención del usuario. En cambio, en imágenes de observatorio cada capa es seleccionada minuciosamente en ciertas bandas del espectro, permitiendo representar la distribución de los distintos elementos los cuales emiten en ese espectro sobre la imagen haciendo uso de los colores asociados a cada canal (ver Figura 2.5).

Para dicha imagen de ejemplo se usa la banda del Hidrógeno para las tonalidades verdes,



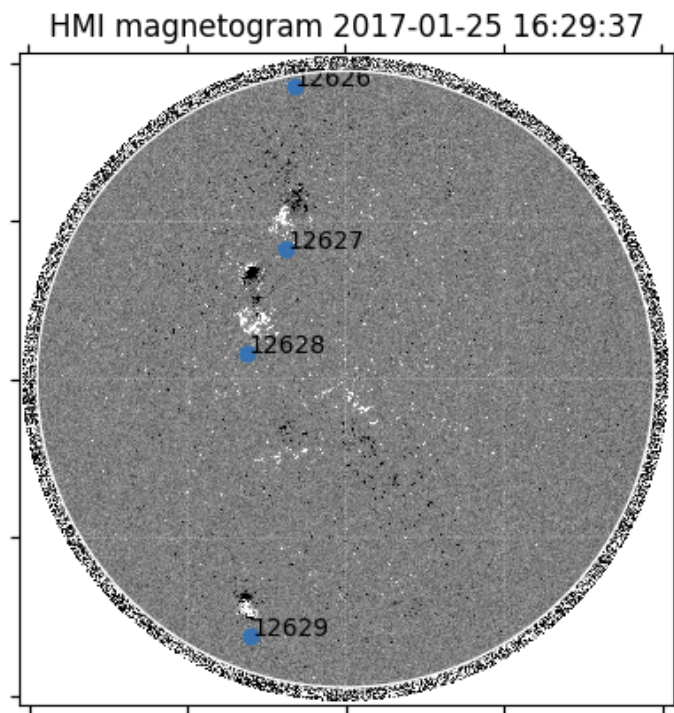


**Figura 2.5:** *Procesamiento de las imágenes de la Nebulosa de la Laguna obtenido con la sonda Hubble.*

la del  $O_3$  para los tonos azulados y el  $N_2$  para los tonos rojos. Cabe destacar que aunque el Nitrógeno y el Hidrógeno se vean con tonalidades rojizas en el espectro visible, le cambiamos el espectro al hidrógeno para poder diferenciar en su conjunto la mezcla de esos elementos en la imagen. Mediante este proceso multibanda, podemos obtener en una imagen una posible distribución de esos tres elementos. Para todo ello, los instrumentos como la sonda Hubble contiene un total de 63 filtros en bandas, desde los *ultravioletas* y *luz visible* (*UVIS*) hasta filtros en el *infrarrojo* (*IR*) que nos permiten crear esas combinaciones. Para ello hace uso de un sensor *CCD* monocromático con una gran resolución ( $2048 \times 4096$  píxeles) en su cámara *WFC3* (*Wide Field Camera 3*).

La necesidad de tantos filtros es necesario para permitir abarcar grandes campos de estudio [8]. Un ejemplo de ello es que ciertas capas o regiones del Sol son muy poco densas (tales sean cromosfera o corona solar) y se necesitan de situaciones especiales para poder ser observadas (por ejemplo, los eclipses solares). Para ciertos estudios se hace uso también

de lecturas magnéticas como usa el instrumento *HMI* mediante el efecto *Zeeman*. Para su aplicabilidad, se hacen barridos con el sensor acoplado, detectando las distintas divisiones de bandas de diferentes elementos en función de cómo de intenso sea el campo magnético en dichas áreas y de su polaridad.



**Figura 2.6:** *Magnetograma del Sol tomado por el instrumento HMI de la sonda SDO el 25 de enero de 2017.*



## Capítulo 3

# Sistemas de coordenadas y orientación de imágenes

A partir del siglo XX con la digitalización de la astronomía y el avance de la ingeniería, se han presentado gran variedad de métodos de representación del Sol acompañados de sistemas de coordenadas. Estos son usualmente expresados mediante longitudes y latitudes en grados y tratan de simular los sistemas de posicionamiento terrestres, su precisión y su flexibilidad. Dichos sistemas requieren de datos muy precisos recogidos de las efemérides, los cuales son sistemas o programas informáticos cuya finalidad es simular y calcular un gran abanico de variables usadas para los sistemas de posición.

Los sistemas de posicionamiento se aplican generalmente para obtener la posición de un punto sobre el espacio tridimensional. Manejar un sistema tridimensional de forma sencilla y rápida puede conllevar una mayor dificultad que hacer uso de una proyección del mismo sistema sobre figuras para poder obtener un sistema bidimensional o mapa. Estas proyecciones no solo se limitan a representar los datos obtenidos del sensor sobre un fragmento de imagen, sino que pueden componerse entre otras proyecciones para formar sistemas de representación múltiple, que nos permitan visualizar de una forma cómoda por ejemplo la superficie a lo largo de una serie de rotaciones, la cual es la finalidad de un mapa sinóptico.

Para poder entender cómo podemos recrear un sistema de coordenadas sobre el Sol y poder aplicarlo a las propias imágenes, lo primero será tratar sobre los sistemas de coordenadas

geocéntricos.

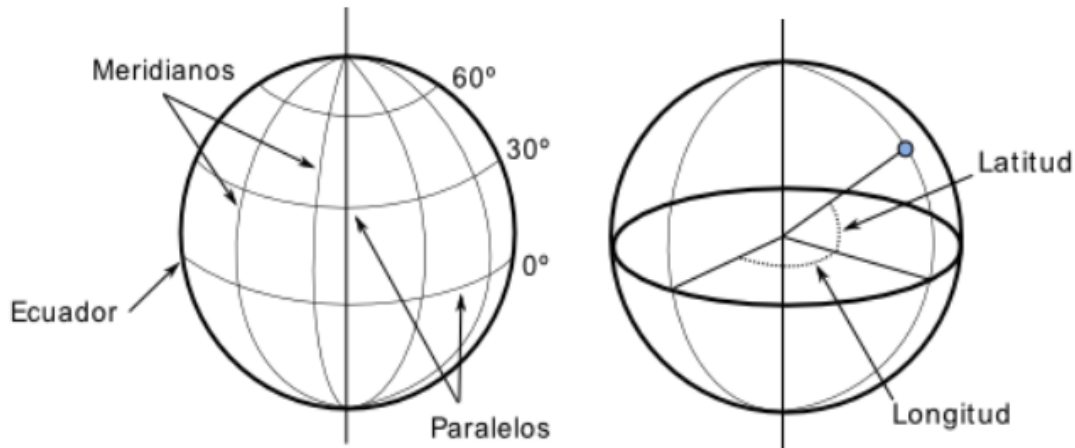
### 3.1. Sistemas de coordenadas geocéntricos

Para poder representar una posición sobre un sistema de coordenadas se hace uso de un sistema de referencia. Los sistemas geocéntricos están descritos sobre cuerpos ideales o perfectos como esferas. Aunque el Sol puede asemejarse a una esfera perfecta vista por el observador, la Tierra en cambio no puede hacerlo a un cuerpo ideal, por ello que se usen sistemas de representación basados en elipsoides o esferoides (achataados por los polos) para la Tierra y esféricos para el Sol.

Debido a que el problema que se quiere brindar durante este trabajo es en el campo solar, trataremos de cómo usar estos sistemas de referencia para posicionarnos sobre el Sol, y poder usarlos para localizar y posicionar elementos tales sean manchas solares o erupciones entre otros.

#### 3.1.1. Sistemas de coordenadas geocéntrico terrestre

El sistema de coordenadas geocéntrico permite representar cualquier punto sobre la superficie de la esfera mediante el uso de dos ángulos y un radio (que indica la altura al punto). En un sistema **terrestre** dichos ángulos se denominarán latitud y longitud. La latitud tomará como referencia el Ecuador y variará entre  $-90^\circ$  a  $90^\circ$  representando un punto sobre una vertical (donde  $<0^\circ$  indica el Sur en valor absoluto y en términos positivos representa el Norte). La longitud tomará como referencia el meridiano de Greenwich, que variará entre  $-180^\circ$  y  $180^\circ$  (para ángulos negativos indicará el Oeste y positivos el Este). Generalmente se usa esta notación, aunque existen variaciones donde el ángulo para representar la vertical varía de 0 a  $180^\circ$  y la horizontal desde 0 a  $360^\circ$ . Denominaremos paralelos a los planos paralelos con el Ecuador y meridianos a los planos paralelos con el meridiano de Greenwich (ver Figura 3.1).



**Figura 3.1:** *Representación de un sistema de coordenadas geocéntricas terrestres.*

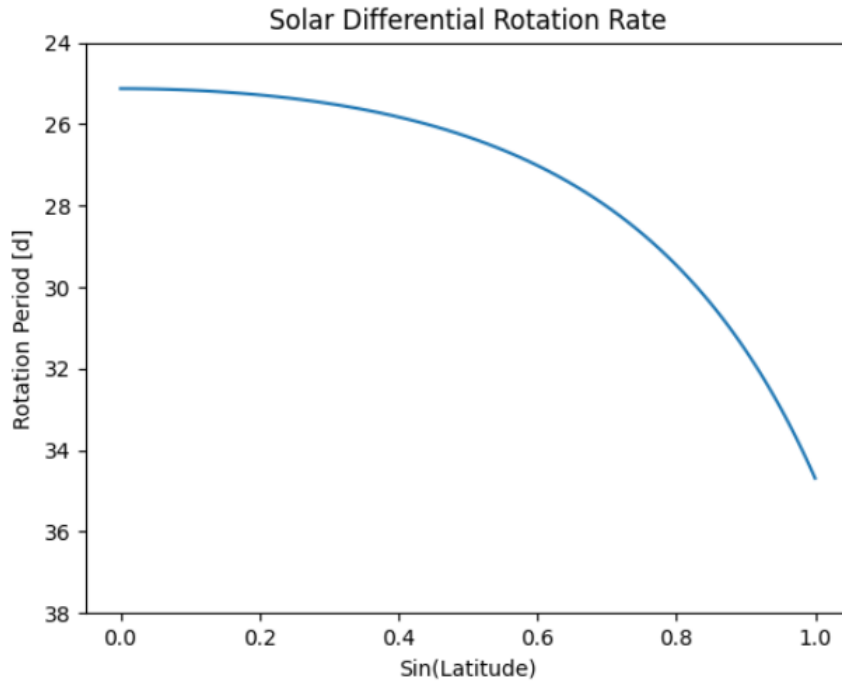
### 3.1.2. Sistemas de coordenadas heliográficas

El sistema de coordenadas heliográficas permite situar cualquier punto sobre la superficie del Sol mediante una latitud y una longitud similar al sistema geográfico, haciendo uso de un sistema de referencia más complejo basado en planos fundamentales de movimientos.

Como el sistema observador será la Tierra y no el propio objeto, deberemos de crear un sistema de referencia en conjunto al observador y el Sol. Este sistema deberá de ser muy preciso ya que si no se hacen correcciones de posicionamiento, el observador verá la partes estudiadas en zonas no correspondientes, esto es debido a que los ejes del Sol y de la Tierra están desalineados. Esto se aplicará en el preprocesado de la imagen por parte del instrumento para orientar de una manera óptima y precisa los datos. Para realizar dicha corrección será necesario saber en qué modo se alinean o desalinean dichos ejes proyectados entre muchas otras variables, haciendo uso de los distintos movimientos que le corresponde al Sol como objeto sobre el que se proyectará el sistema de coordenadas, y la Tierra como observador [9, 10]. Dichos movimientos son:

- El **movimiento de rotación** es el giro de un cuerpo sobre su propio eje (pasa por los

polos) y va en función de la velocidad angular. En el caso de la Tierra, el periodo de rotación es de 23 horas y 56 minutos. En cuerpos gaseosos como ocurre en el Sol, esta velocidad de rotación cambia según su latitud, por ello que se denomine movimiento de rotación diferencial [11]. Esto nos limita a la hora de hacer cálculos, debido a que el periodo no es el mismo para cada latitud (ver Figura 3.2). En el Ecuador solar, el periodo se aproxima a 24.5 días y en latitudes cercanas a los polos, alrededor de 34 días.



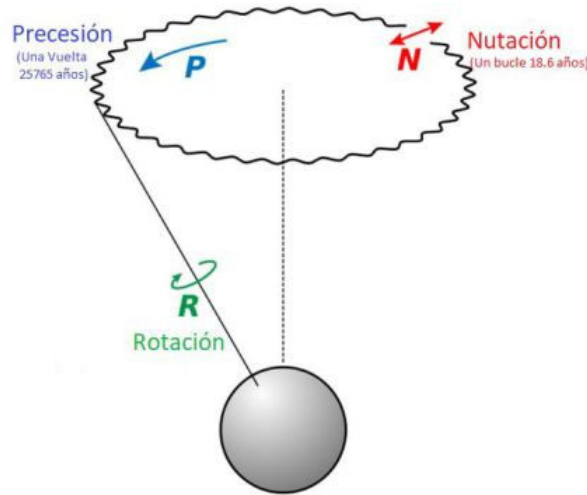
**Figura 3.2:** Representación de la velocidad angular en función de la latitud en valor absoluto.

Para el cálculo de dichos valores se han utilizado lecturas recopiladas durante décadas de posicionamiento de manchas solares y demás estructuras observables en el Sol, para poder medir con una debida precisión este valor en función de la latitud. Es importante hacer relevancia a este movimiento, debido a que no es propio de cuerpos que se comportan como un cuerpo sólido.

- El **movimiento de traslación** es aquel que realiza un cuerpo respecto a otro. En

el caso del sistema solar, todos los elementos que orbitan al Sol se moverán en una elipse en la que el Sol ocupará uno de sus focos. El plano sobre el que se realiza dicho movimiento se denomina eclíptica y a la normal del plano se le denomina eje de la eclíptica. Debido a las características de la órbita que siguen los cuerpos en rotación, esta geometría permite proyectar los cuerpos entre ellos o sobre un plano en el espacio.

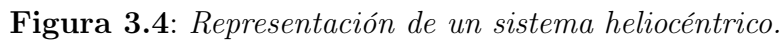
- El **movimiento de nutación y precesión** es aquel que posee cuerpos en rotación tales como planetas, peonzas y giroscopios entre muchos otros, donde el objeto gira sobre un eje de rotación que pasa por sus polos, y a su vez, este eje gira sobre un punto dado (ver Figura 3.3). El ángulo que forma el eje de rotación de un cuerpo respecto al eje de la eclíptica se denomina oblicuidad.



**Figura 3.3:** Representación de los movimientos de precesión y nutación.

Para el desarrollo de este sistema de coordenadas sobre el Sol, lo primero será crear la proyección del Sol y la eclíptica sobre un plano (ver Figura 3.4). Sobre dicha proyección, denotaremos diferentes puntos que serán usados por nuestro sistema de coordenadas helio-gráfico y calculados por las efemérides haciendo uso de trigonometría esférica. Entre sus principales elementos de posicionamiento nos encontramos:





- Tras todo ello, podemos representar cualquier punto de la superficie solar mediante dos ángulos,  $L$  y  $B$ .  $L$  se usará para representar la longitud y dependerá del punto  $O$  tal se

observa en la imagen y  $B$  para representar la latitud de un punto referenciado desde el Ecuador solar. La longitud tomará valores entre 0 a  $360^\circ$  y la latitud entre  $\pm 90^\circ$ .

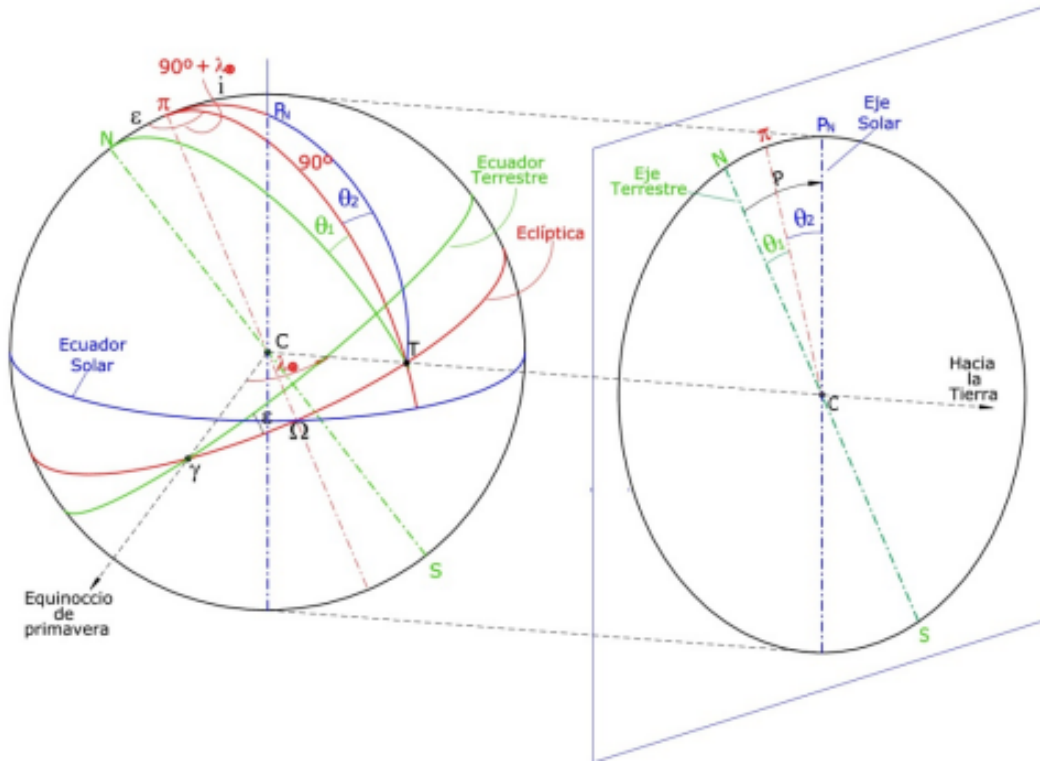
## 3.2. Proyecciones de sistemas de coordenadas

Al tomar una imagen, ya sea por un telescopio, un instrumento en el espacio o a simple vista, se está representando el sistema de cierta manera sobre un plano. Un plano es una representación de un sistema tras proyectarlo sobre dos dimensiones. Existen diversos métodos de proyección, y esos varían según la finalidad. Entre tipos de proyecciones pueden encontrarse proyecciones cilíndricas, cónicas y cenitales entre otras, dependiendo del cuerpo que se use para proyectar. Nos centraremos en este capítulo sobre todo en la proyección cenital y la cilíndrica.

Para crear una proyección cenital u ortonormal de una esfera, tendremos que proyectarla sobre un plano. Cuando tomamos una imagen a través de un satélite o una sonda, podemos apreciar en el caso de la Tierra, la esfera desde un lateral visualizándola como un círculo, el Ecuador centrado como una línea recta y los polos en ambos extremos (superior e inferior). En cambio, cuando queremos representar una superficie sobre un plano como si de una carta náutica se tratase, haremos uso de una proyección cilíndrica. Como ejemplo para su visualización, pondremos el planeta Tierra en el interior de un cilindro, de radio el terrestre. Al proyectarlo sobre las paredes interiores del cilindro, nos quedará una representación similar al clásico mapamundi. Existen gran variedad de sistemas de ecuaciones que intentan emular esta proyección, de entre esos sistemas se puede enumerar por ejemplo: la proyección Mercator, Peters, etc. A lo largo de esta sección, se explicará cómo se plantea este método para el Sol, y cómo afecta estas proyecciones a los sistemas de coordenadas aplicados. Esta planificación es importante, ya que permitirá posicionar la imagen en su debida posición para que las manchas y demás estructuras se vean moverse en un sentido lineal y poder posicionar dichos elementos de una forma precisa y óptima.

### 3.2.1. Proyección cenital de un sistema tridimensional heliográfico

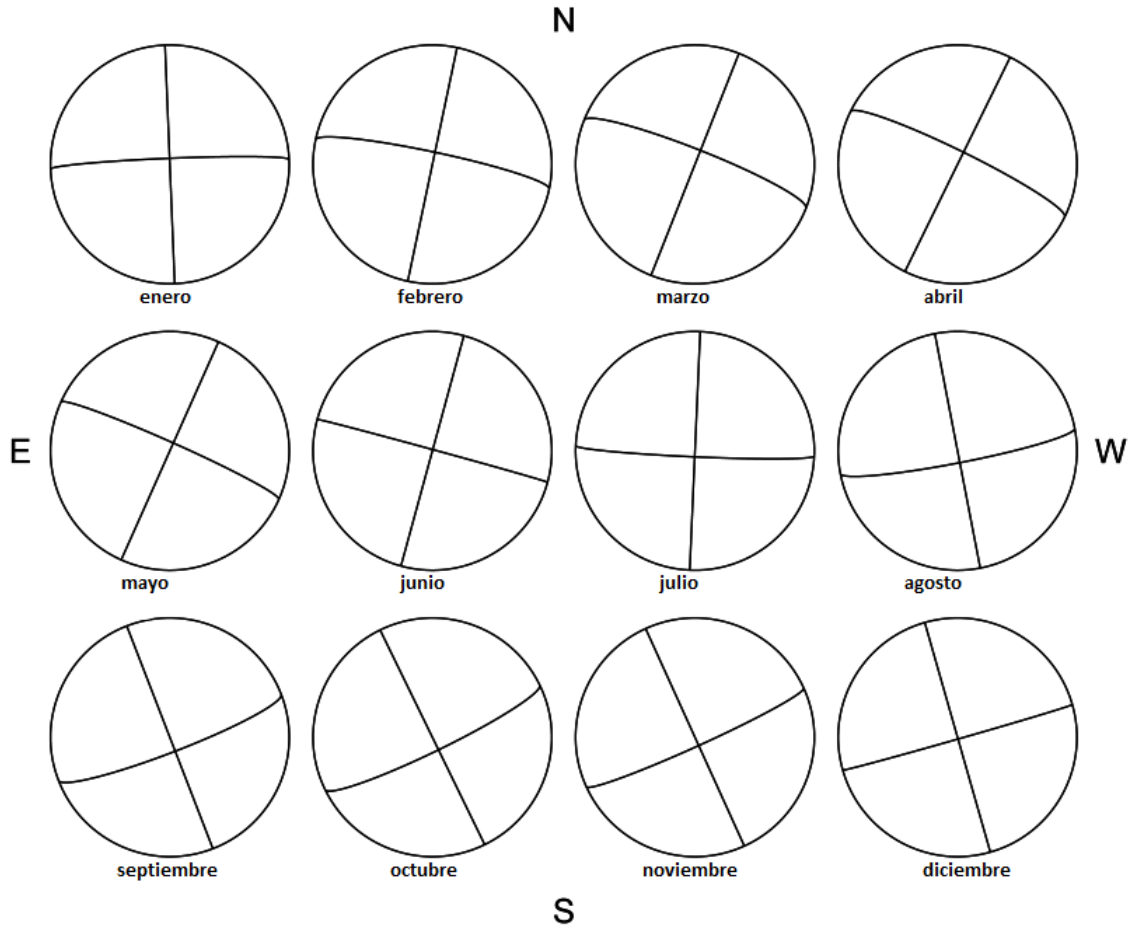
Anteriormente se ha explicado cómo calcular la posición de un punto en la esfera del Sol mediante dos ángulos  $B$  y  $L$  que dependerá de otros puntos cuyos valores nos son dados por las efemérides [12]. Tal como se ha introducido en el apartado anterior, se tendrán que proyectar los ejes terrestres sobre el sistema creado anteriormente (ver Figura 3.4) obteniendo el siguiente sistema representado en la Figura 3.5.



**Figura 3.5:** *Proyección de un sistema de coordenadas heliográficas sobre un plano.*

En la imagen proyectada,  $\pi$  será la normal al plano de la eclíptica,  $P_n$  el eje de rotación solar y  $N$  el eje de rotación terrestre. Entre los ángulos observables tenemos  $\theta_1$  que representa la oblicuidad del eje terrestre y  $\theta_2$  la oblicuidad del Sol. Con ellos podemos calcular el ángulo  $P$  cuyo valor está en el intervalo  $\pm 23.5^\circ$ , el cual nos brinda información sobre como de inclinado está el eje de rotación del Sol respecto a la Tierra, de izquierda a derecha. Debido

a la geometría de la órbita Sol-Tierra existe un movimiento aparente hacia los lados y a su vez, otro movimiento de dicho eje más cerca o más lejos del observador (hacia delante y atrás). Dicha inclinación varía entre  $\pm 7.25^\circ$  (a este ángulo se le denominará  $B_0$ ). El ángulo  $B_0$  y  $P$  son los valores que necesitamos para poder sacar las coordenadas en un punto obteniéndolas de las efemérides. La variación de estos valores puede verse gráficamente en la Figura 3.6.

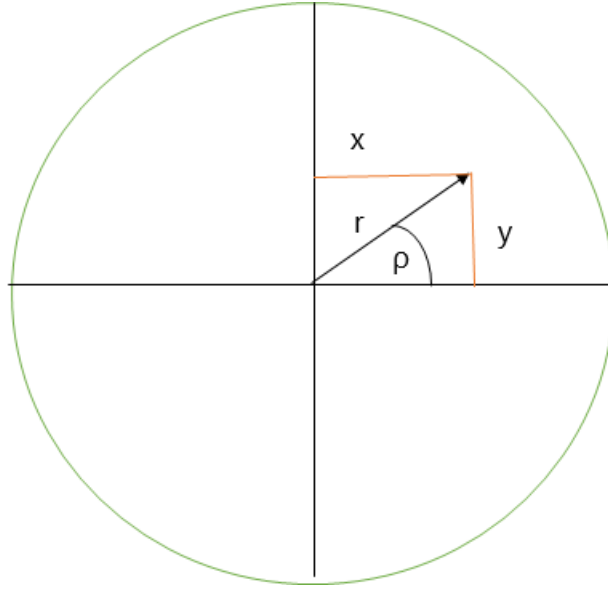


**Figura 3.6:** *Variación de los ángulos de inclinación del Sol.*

Una vez tenemos los parámetros del sistema para poder representar cada uno de sus ejes para posicionar un elemento, ahora deberemos de crear en sí el sistema de ecuaciones sobre dichos parámetros [13]. Existen diferentes formas para calcularlas:

## 1) Coordenadas polares

Debido a que en la proyección que nos brinda la imagen es un círculo, podemos aplicar coordenadas polares respecto al eje N-S terrestre para obtener unas coordenadas del punto en función de un radio y un ángulo, y a través de estos dos valores, calcular su longitud  $L$  y latitud  $B$  en grados. Una representación de este sistema puede observarse en la Figura 3.7.



**Figura 3.7:** *Coordenadas polares.*

El primer paso para la obtención de coordenadas será calcular el radio  $r$  o distancia al punto, tomando como origen el centro del disco solar aparente, mediante dos posiciones,  $x$  e  $y$ .

$$r = \sqrt{x^2 + y^2} \quad (3.1)$$

El siguiente paso será calcular el ángulo que forma el vector  $r$  con el eje  $x$ :

$$\rho = \tan^{-1}\left(\frac{y}{x}\right) - P \quad (3.2)$$

Para reducir las siguientes ecuaciones de posicionamiento se hará uso del siguiente parámetro sigma. Para calcularlo se hará uso del radio  $r$  que irá desde el centro del disco solar observable hasta el punto a posicionar y se usará también el radio del disco solar  $R$  medido

sobre la imagen.

$$\sigma = \sin^{-1}\left(\frac{r}{R}\right) \quad (3.3)$$

Una vez obtenidos los datos y las fórmulas de longitud y latitud en función de  $B_0$ ,  $\sigma$  y  $\rho$ , pueden calcularse los valores en grados. Dichas fórmulas son:

Para la latitud:

$$B = \sin^{-1}(\cos\sigma \cdot \sin B_0 + \sin\sigma \cdot \cos B_0 \cdot \sin\rho) \quad (3.4)$$

Para la longitud:

$$L = \sin^{-1}\left(\frac{\cos\rho \cdot \sin\sigma}{\cos B}\right) \quad (3.5)$$

## 2) Plantillas Stonyhurst

Muchos observadores amateurs pueden no tener una visión matemática o simplemente se quiera hacer un posicionamiento más rápido sobre la imagen, por ello que hagamos alusión al siguiente método haciendo uso de plantillas [14]. Las plantillas son conocidas como plantillas Stonyhurst, las cuales serán posicionadas en función de los ángulos  $B_0$  y  $P$  que obtendremos de las efemérides. Estas plantillas funcionan exactamente igual a la hora de posicionar que las coordenadas geográficas, mediante longitudes y latitudes.

Lo primero que hay que tener en cuenta es el ángulo  $P$ , situando los ejes en función de este. Para ello existe el disco de Porter (las líneas que representa milímetros en los dos laterales de la Figura 3.8), el cual usaremos como guía para situar el eje E-O o meridiano solar. Una vez hecho esto, sacamos su eje vertical N-S solar mediante una mediatriz. Tras situar el eje horizontal y vertical, deberemos de hacer uso de una plantilla que varía en función del ángulo  $B_0$ . Esto es debido a que dicho ángulo nos indica como varía la posición del eje de rotación hacia delante y hacia atrás. Un ejemplo con  $P$  de  $14.30^\circ$  y una  $B$  de  $6.46^\circ$  es la mostrada en la Figura 3.8.

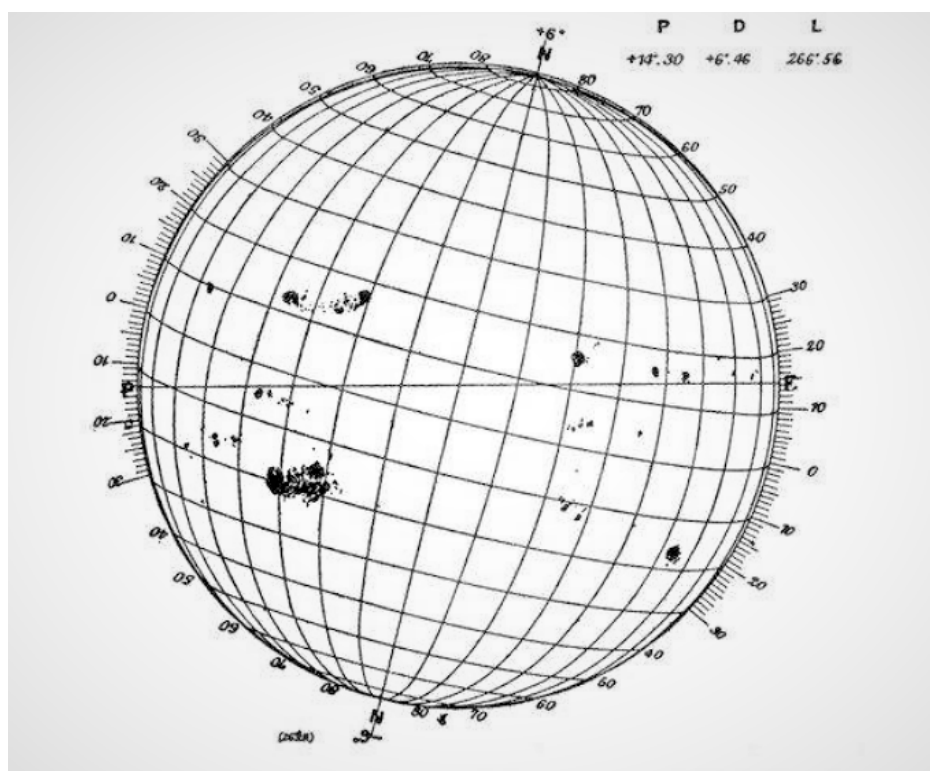


Figura 3.8: Método de posicionamiento mediante uso de plantillas.

# Capítulo 4

## Mapas sinópticos

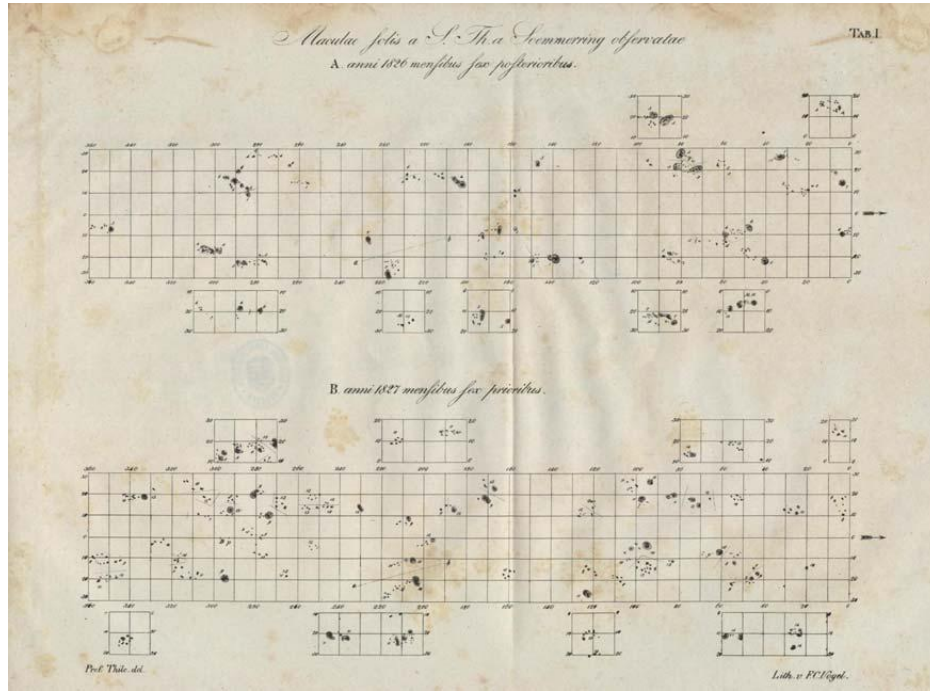
Las primeras representaciones de los mapas sinópticos en la historia solar se han hecho bajo el nombre de diagramas de *Thilo* (ver Figura 4.1) publicados en 1828 por L. Thilo en una de sus disertaciones. En ellos se trazan los puntos de observación (en su mayoría manchas solares), a lo largo de una serie de coordenadas (originalmente en longitudes y latitudes) sobre líneas verticales y horizontales [15]. La principal finalidad de estos mapas es lograr representar el sistema a lo largo de una serie de rotaciones de una superficie esférica, pudiéndose aplicar esta representación no únicamente al Sol, sino a más representaciones como la Tierra o Marte. Se hará uso de diferentes figuras de la Tierra para tener una visión más clara al proceso de proyección de un sistema a otro, aunque es igualmente válido para el Sol.

Para realizar dichos mapas, el primer paso será obtener las imágenes del sensor. Estas imágenes, en función del usuario y el instrumento, serán sometidas a un preprocesado inicial como ajuste de profundidad, ecualizaciones, etcétera. Una vez hecho este paso, se orientará la imagen haciendo uso de los sistemas planteados en el anterior capítulo y se proyectará sobre un sistema bidimensional. Para dicha proyección, haremos uso de la representación *squircle* <sup>1</sup>.

---

<sup>1</sup><https://pypi.org/project/squircle/>





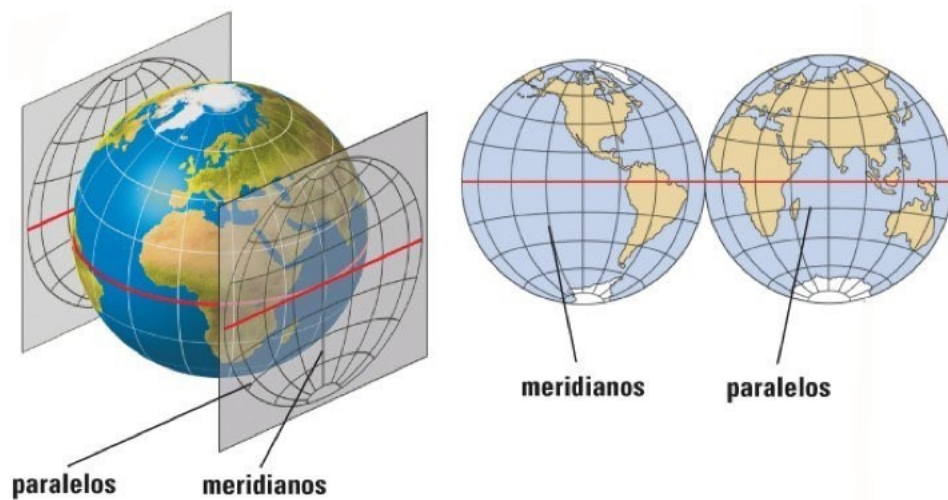
**Figura 4.1:** Representación de rotaciones solares mediante el diagrama de Thilo.

## 4.1. Proyección a coordenada cilíndrica

Cada imagen en proyección cenital brinda  $180^\circ$  de la superficie proyectada sobre un plano. Si tomamos otra imagen de la parte posterior desde otro observador, podríamos representar toda la superficie mediante dos proyecciones cenitales (ver Figura 4.2).

Para poder realizar esta proyección de forma sencilla, se deberá crear una expansión de la imagen en proyección cenital para lograr una versión rectangular de dicho círculo proyectado. Para ello, haremos uso de un sistema basado en la figura geométrica *squircle* (también conocida como círculo truncado). Una *squircle* es una figura entre un círculo y un cuadrado, la cual haciendo uso de sus fórmulas trataremos de expandir la imagen (ver Figura 4.3).

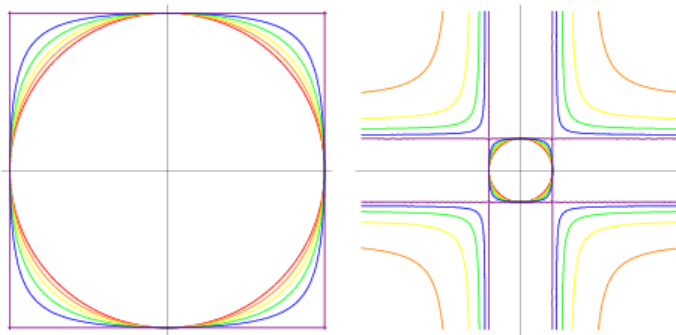
Existen distintas aplicaciones de este método y sus ecuaciones son variadas con tal de conseguir la mejor aproximación posible. Entre los métodos que nos brinda la librería podemos encontrar las tres siguientes.



**Figura 4.2:** *Representación de la superficie terrestre mediante dos proyecciones cenitales.*

#### 4.1.1. Fernández Guasti Squire

Este método es conocido como squire FGS, cuya apariencia es la siguiente:



**Figura 4.3:** *Proceso FGS.*

Esta transformación sobre la imagen busca lograr una figura que haga de paso entre un círculo y un cuadrado de manera uniforme, reduciendo el máximo el error de proyección.

Sus ecuaciones son las siguientes:

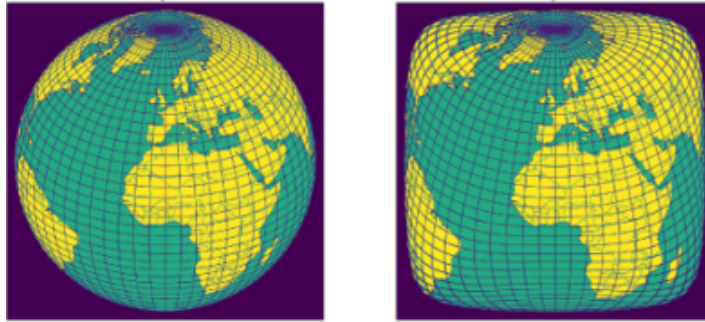
$$x^2 + y^2 - \frac{s^2}{r^2} \cdot x^2 y^2 = r^2 \quad (4.1)$$

$$0 \leq s \leq 1 \quad (4.2)$$

Cuando  $s$  es 0, las ecuaciones representan un círculo de radio  $r$  y cuando  $s$  es 1, es un cuadrado de lado  $r$ . Tal y como podemos ver en la Figura 4.3, para tomar la parte central delimitaremos con las siguientes cotas:

$$-r \leq x \leq r, -r \leq y \leq r \quad (4.3)$$

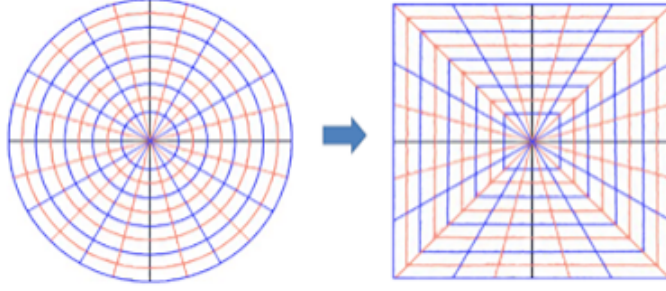
El resultado del proceso sobre una imagen junto a sus meridianos puede observarse en la Figura 4.4. En ella se puede encontrar a su izquierda la imagen original y a su derecha, la aplicación de la transformación.



**Figura 4.4:** *Método FGS sobre meridianos.*

### 4.1.2. Simple Stretching

Se busca realizar una expansión lineal de forma radial, haciendo uso de los radios del círculo inicial y expandiendo en función de estos. Es un método útil, aunque sus ecuaciones son de mayor dificultad en su aplicabilidad. El resultado general que se quiere obtener es el siguiente (ver Figura 4.5):



**Figura 4.5:** *Proceso simple stretchng.*

Para hacer uso de esta transformación, lo primero es tener en cuenta dos cosas. La primera de ellas es que el par de variables  $(u, v)$  corresponde a los valores de un círculo de radio unidad y situado en el punto  $(0, 0)$ . Lo segundo es que las variables  $(x, y)$  denotan los puntos correspondientes en un rectángulo unidad de lado 1 y situado también en el punto  $(0, 0)$  en los ejes. En función de  $u$  y  $v$ :

Siempre que  $u^2 \geq v^2$ :

$$x = \text{signo}(u) \cdot \sqrt{u^2 + v^2} \quad (4.4)$$

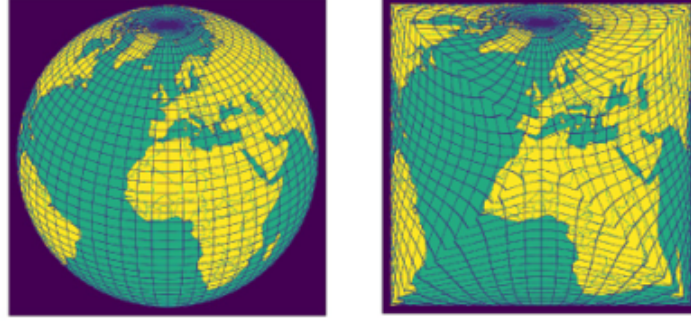
$$y = \text{signo}(u) \frac{v}{u} \sqrt{u^2 + v^2} \quad (4.5)$$

En caso contrario:

$$x = \text{signo}(v) \frac{u}{v} \sqrt{u^2 + v^2} \quad (4.6)$$

$$y = \text{signo}(v) \sqrt{u^2 + v^2} \quad (4.7)$$

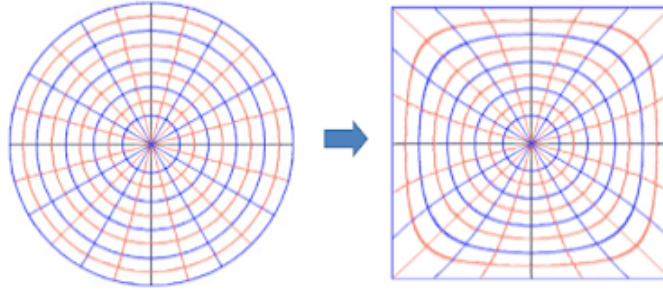
El resultado de este método al realizar las transformaciones de los puntos sobre escalas regulares y radiales, presentará bordes y puntos visibles sobre la proyección quedando la representación como la siguiente imagen (ver Figura 4.6):



**Figura 4.6:** *Proceso simple stretching sobre meridianos.*

### 4.1.3. Elliptical Grid Mapping

Es un método similar al anterior, pero intentando depositar los elementos de la imagen original en la proyectada de forma no tan lineal, debido a que como se comprueba en la imagen derecha de la Figura 4.7, quedan picos o cambios bruscos visibles una vez realizada la transformación. El resultado es el siguiente:



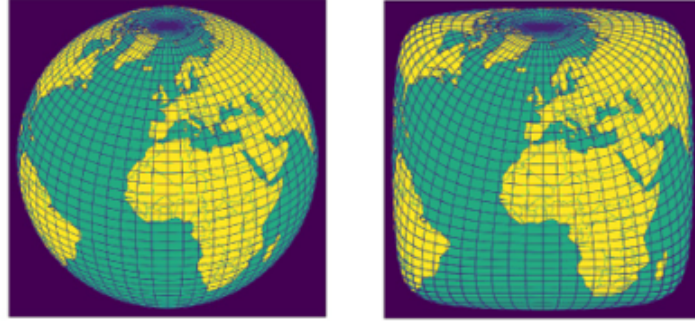
**Figura 4.7:** *Proceso elliptical squircle.*

Sus ecuaciones harán uso de la misma representación de coordenadas que el método *Simple Stretching*, los puntos  $(u, v)$  para situar los puntos en el disco y los puntos  $(x, y)$  para los puntos del cuadrado. Dichas ecuaciones son:

$$x = \frac{\sqrt{2 + 2u\sqrt{2} + u^2 - v^2}}{2} - \frac{\sqrt{2 - 2u\sqrt{2} + u^2 - v^2}}{2} \quad (4.8)$$

$$y = \frac{\sqrt{2 + 2v\sqrt{2} - u^2 + v^2}}{2} - \frac{\sqrt{2 - 2v\sqrt{2} - u^2 + v^2}}{2} \quad (4.9)$$

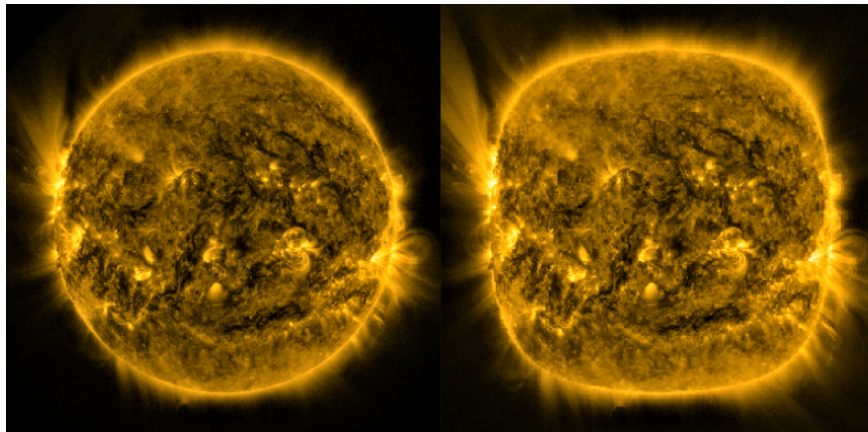
Como resultado final de este método puede observarse que se han eliminado dichos picos o relieves que no interesan tener, como puede apreciarse en la Figura 4.8.



**Figura 4.8:** *Proceso elliptical squircle sobre meridianos.*

Tras mostrar los tres métodos, vemos que los más viables son el *FGS* (que corresponde al primer método explicado) y el método *Elliptical Grid Mapping* (que corresponde con el último). Como la ecuación del primer elemento es mucho más sencilla de aplicar que la del último y a su vez, esta deposita los elementos a la hora de hacer la transformación de una forma más estable, brindará de un menor error dándole uso al primer método *FGS*.

Al aplicar dicha transformación *FGS* sobre el Sol, se mostrará el siguiente resultado (ver Figura 4.9):

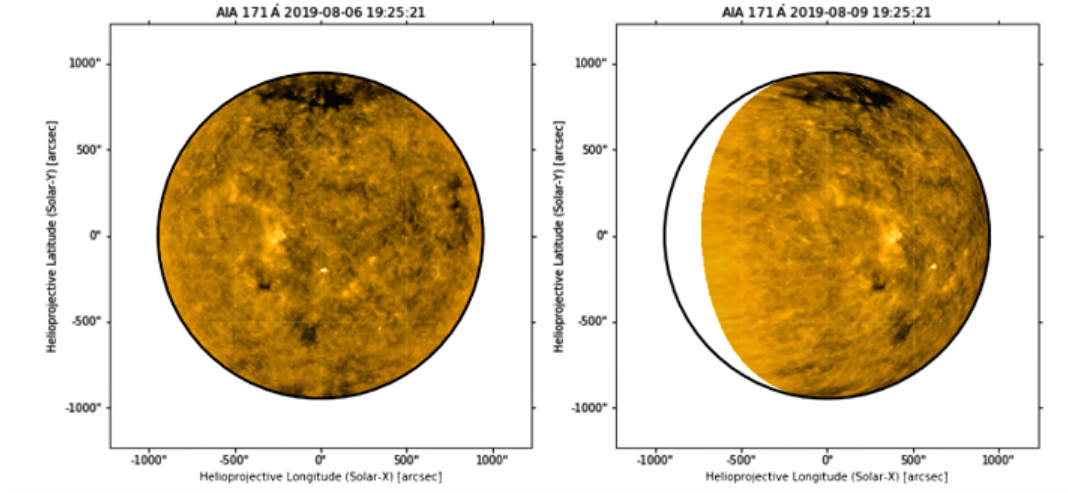


**Figura 4.9:** *Aplicación del método FGS a una imagen solar real.*



## 4.2. Fragmentación de imágenes

Cuando se tienen dos tomas del Sol en un cierto lapso de tiempo, parte de la imagen menos reciente estará presente en la superficie visible de la imagen más reciente (debido a la propia rotación del Sol tal y como puede observarse en la Figura 4.10). Este solapado es un problema debido a que no puede aparecer píxeles repetidos en distintas zonas sobre el mapa sinóptico. Por ello que se deba calcular cuantos píxeles se requieren en función de dicho lapso de tiempo.



**Figura 4.10:** *Simulación de la rotación diferencial para 4 días.*

Para dicho cálculo, el primer paso será determinar la diferencia de tiempo entre toma y toma. Dicho lapso será descrito mediante:

$$\Delta t = t_a - t_{ant} \quad (4.10)$$

Tomando como  $t_a$  la fecha de la imagen más reciente y  $t_{ant}$  como la fecha de la imagen menos reciente, y la diferencia de ambas será el incremento. Una vez calculado dicho lapso, el siguiente paso será calcular qué proporción de la imagen se debe tomar.

La rotación solar depende directamente de la velocidad angular del Sol. Dicha velocidad, como varía en función de la latitud  $\phi$  en la que se mida, se denominará velocidad angular diferencial o de forma general, rotación diferencial. Dicha rotación se describe de la forma siguiente:

$$w = A + B\sin^2\phi + C\sin^4\phi [^\circ/\text{day}] \quad (4.11)$$

$$A = 14,713 \pm 0,0491 [^\circ/\text{day}] \quad (4.12)$$

$$B = -2,396 \pm 0,188 [^\circ/\text{day}] \quad (4.13)$$

$$C = -1,787 \pm 0,253 [^\circ/\text{day}] \quad (4.14)$$

Una vez declarada la velocidad angular en función de la latitud, el siguiente paso será obtener el periodo de rotación en función de la latitud, por ello:

$$w = \lim_{\Delta t \rightarrow 0} \frac{\Delta\phi}{\Delta t} = \frac{360}{T} [^\circ/\text{day}] \quad (4.15)$$

$$T(\phi) = \frac{360^\circ}{w^\circ/\text{day}} = \frac{360^\circ}{A + B\sin^2\phi + C\sin^4\phi [^\circ/\text{day}]} \quad (4.16)$$

Como se tiene calculado el periodo en función de la latitud, se puede hacer uso de él para poder calcular cuantos grados varía dado un incremento de tiempo (dado en días). Esta variación en grados al fin y al cabo, al depender el periodo de la latitud, también dependerá de la latitud. Se representará mediante  $\xi(\phi)$  y se calcula de la siguiente manera:

$$\xi(\phi) = \frac{360 \cdot \Delta t}{T(\phi)} [^\circ] \quad (4.17)$$

Debido a que el manejo de las imágenes se realiza mediante píxeles en todo momento, el siguiente paso será describir a cuantos píxeles equivale la diferencia de ángulos calculados en la ecuación anterior en función de la latitud. En las imágenes solares, el Sol se representa en una proyección cenital de  $180^\circ$  de superficie. Si esos  $180^\circ$  son divididos entre el diámetro



observable en píxeles del Sol, se podrá calcular el factor de escala en  $^{\circ}/\text{píxel}$  representado con  $\delta$ , para su cálculo:

$$\delta = \frac{180}{2 \cdot R_{Sol}} [^{\circ}/\text{pixel}] \quad (4.18)$$

Una vez calculado estos dos últimos factores, se puede corresponder cuantos píxeles equivale la diferencia de ángulos en función la latitud.

$$\lambda(\phi) = \frac{\xi(\phi)}{\delta} [^{\circ}] \quad (4.19)$$

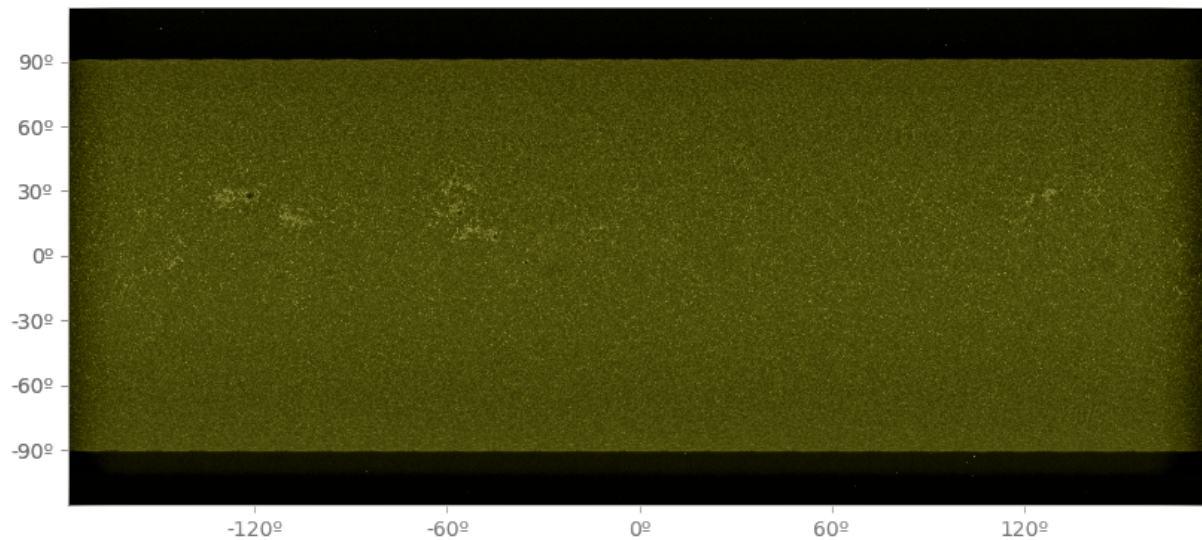
La mayoría de las veces, los instrumentos solares almacenan los datos en la cabecera *FITS* en unidades de arcosegundo en vez de en grados o en píxeles. Las unidades se especifican en la tarjeta *CUNIT* de la cabecera. La cabecera *FITS* también suele traer un factor de escala en unidades de arcosegundos por píxel que puede encontrarse en la tarjeta *CRPIX<sub>i</sub>*, siendo *i* el factor de escala para cada eje. A su vez, se puede encontrar el radio del Sol en unidades de arcosegundo en la cabecera con la tarjeta *RSUN*. Con todo ello, se puede realizar la conversión de grados por píxel a grados por arcosegundos mediante el campo *CRPIX<sub>i</sub>*.

Cuando se hace uso de las proyecciones cilíndricas, los puntos se expanden usualmente hacia la derecha e izquierda en función de la posición en la que se encuentre dicho punto. La rotación diferencial en una imagen expandida, por lo tanto, se reduce en su mayor parte a una rotación constante que no depende de la latitud. La composición, una vez fragmentada cada imagen será mucho más sencilla de realizar.

Dicho periodo constante se representa como periodo de Carrington, en el que un mapa sinóptico representará  $360^{\circ}$  de longitud cuando se componen 27.2753 días en imágenes. Por tanto, se declarará la rotación de Carrington como un número entero que representa 27.2753 días.

### 4.3. Composición y validación

Una vez se tienen todos los fragmentos, como último paso, se tendrán que juntar cada uno de forma horizontal en orden cronológico. Haciendo uso de este método para realizar un mapa sinóptico sobre las tomas del instrumento SDO/AIA sobre una longitud de onda de 1600 Ångstrom, desde el día 1 hasta el 28 de enero de 2017, el resultado puede visualizarse en la Figura 4.11. Tal y como representa los diagramas de Thilo introducidos anteriormente, para posicionar un punto sobre el mapa, se ha hecho uso de una escala en grados.



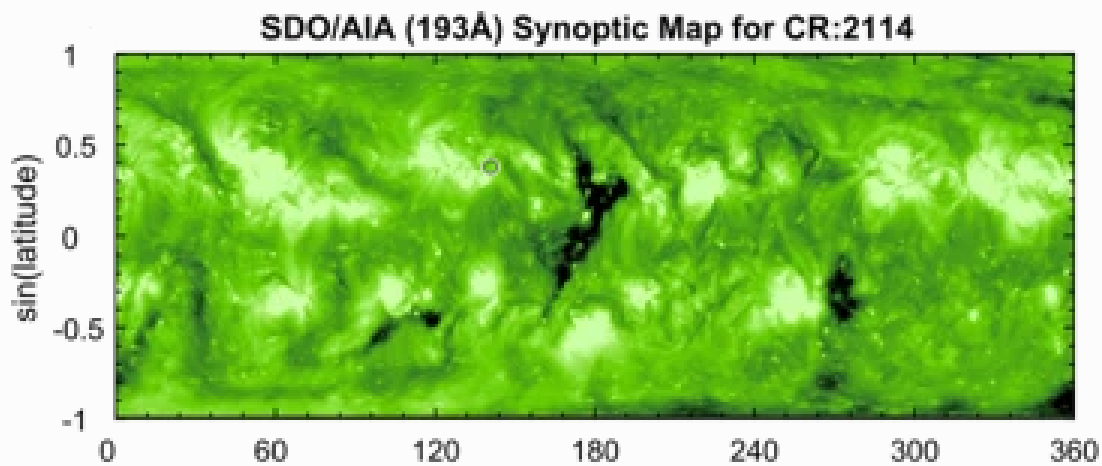
**Figura 4.11:** *Composición de un mapa sinóptico a través de una secuencia de imágenes comprendidas entre el 1 al 28 de enero de 2017.*

Para la **fase de validación del mapa**, se han realizado una serie de comparaciones con resultados de algunos observatorios y expertos en un mismo lapso de tiempo. Como experto se hará hincapié en los resultados obtenidos por el astrónomo A.Hamada [16]. Debido a que existen poca variedad de los mismos, este proceso ha sido difícil de realizar. A este problema se le suma que estos mapas tienen diferentes formas de ser realizados, eligiendo cada observatorio un tipo distinto de proyección y pseudocolor, variando en cierto grado los resultados y los distintos mapas<sup>2</sup>.

---

<sup>2</sup><https://sdo.gsfc.nasa.gov/data/>

Un ejemplo de ello es el siguiente mapa sinóptico sobre la rotación de Carrington 2114, correspondiente al intervalo de tiempo desde el día 26 de agosto de 2011 hasta el día 22 de septiembre de ese mismo año. Este mapa fue realizado por un experto, dando como resultado el encontrado en la Figura 4.12.

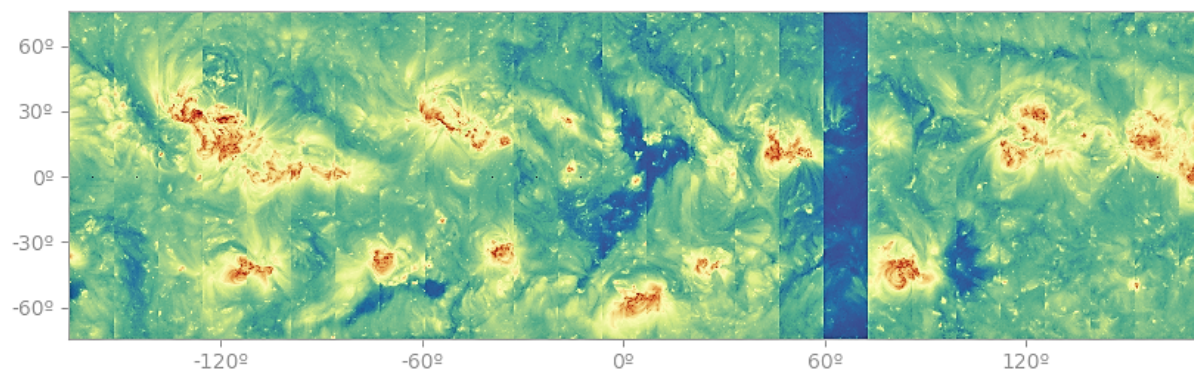


**Figura 4.12:** *Mapa sinóptico correspondiente a la rotación de Carrington 2114 realizado por A.Hamada.*

La imagen se ve bastante limpia de errores debido a que se le ha aplicado un filtro de suavizado para reducir el error de las juntas entre dos tomas distintas de imágenes. Este tipo de imágenes en esta longitud de onda tiene un gran número de detalles, pudiéndose observar los distintos arcos que surgen de la cromosfera y parte de la corona si no se realizan los procesos de suavizado. El resultado de la aplicación en ese mismo lapso de tiempo, sin hacer uso de un filtro de suavizado (aunque la aplicación disponga de él), y en la misma longitud de onda es encontrado en la Figura 4.13.

Aunque el pseudocolor sea distinto entre ambas imágenes, debido a que su uso depende del tipo de estudio que se le quiera realizar al mapa, en el caso del programa se ha escogido uno capaz de resaltar los elementos más luminosos representándolos con tonalidades rojizas. En torno a los 60°, puede observarse una franja atípica sobre la imagen, esto es debido a que el sensor no ha podido recoger la suficiente luminosidad sobre ella.

Respecto a los ejes, este tipo de mapas se representan en grados, aunque es usual verlos



**Figura 4.13:** *Mapa sinóptico correspondiente al realizado por la aplicación.*

también en arcosegundos. En función de cuantas rotaciones se representen, cuando únicamente se visualiza una rotación, el 0 es centrado y cuando hay más rotaciones, el 0 se mantiene a la izquierda.



# Capítulo 5

## Desarrollo de la aplicación

### 5.1. Planificación del desarrollo

Para planificar el proyecto de programación asociado a la memoria, el primer paso será definir el lenguaje de programación a usar, el entorno de desarrollo, los paquetes de los que se hará uso y, sobre qué pilares se apoyará la aplicación.

#### 5.1.1. Lenguaje de programación

Existen diversos lenguajes usados en el ámbito espacial. Este entorno requiere de precisión, de actualizaciones, de una gran comunidad que permita que dicho lenguaje esté en continuo avance, de facilidad de uso y de su posible exportación a distintas plataformas. Existen diversos lenguajes que entran entre esas características como puede ser *C++* o *Java* aunque el que está obteniendo un mayor auge es Python. Entre las características que nos ofrece *Python* destacan:

- **Lenguaje multiparadigma:** *Python* permite hacer uso y control de diferentes técnicas de programación, como puede ser la programación imperativa, declarativa, orientada a objetos o por control de eventos entre otros.
- **Lenguaje interpretado:** *Python* contiene un intérprete, el cual se encarga de traducir el código y ejecutarlo a tiempo real usualmente por un sistema de línea a línea o por un sistema de bloques.

- **Lenguaje multiplataforma:** *Python* contiene una serie de intérpretes los cuales dependen directamente del sistema operativo o la plataforma. Es por ello que el código que se realiza, al no depender del intérprete, no depende tampoco de la plataforma.
- **Lenguaje con tipado dinámico:** Todo lenguaje de programación se fundamenta en tipos de datos o de variables. Un lenguaje será altamente tipado siempre que no se permita tomar valores de diferentes tipos en una única variable. Cuando si lo permitan, y lo hagan en tiempo de ejecución, se denominará lenguaje con tipado dinámico.
- **Flexibilidad:** Debido a que hace uso de un intérprete y de un tipado dinámico, permite que se puedan realizar modificaciones sobre el código sin tener que interpretarlo entero. A su vez, no solo posee flexibilidad en tiempo de ejecución, sino también en módulos. Puede adaptarse un módulo de funciones en C++ u otros lenguajes, y que puedan ser usados por *Python*.
- **Comunidad:** *Python* posee una gran comunidad, no solamente asociada al propio código, sino asociada a numerosas librerías. Esto ha permitido que un mayor número de personas escojan este lenguaje en lugar de otros, permitiendo el avance de un mayor número de líneas de código sobre un gran número de áreas. Un ejemplo de ello es *SunPy*, una comunidad activa encargada sobre todo en crear código para la programación solar.
- **Simplicidad y legibilidad:** *Python* fue creado para brindar un código limpio y fácil de leer. Al no tener que estar pensando en la tipología de las diferentes variables o en como estructurar de forma compleja el código (ya que *Python* está exento de corchetes además de puntos y comas entre otras dificultades) nos permite una estructura simple y fácil de entender.

Entre todo lo anterior cabe destacar un punto muy importante, *Python* es un lenguaje interpretado. Esto tiene sus ventajas pero también sus grandes desventajas, su principal

es que un lenguaje interpretado siempre va a ser más lento que un lenguaje compilado. El proceso de paso a lenguaje máquina en un lenguaje interpretado se realiza cada vez que se ejecuta el código sobre un intérprete, en cambio, cuando haces uso de un compilador, se compila una única vez, y en pasos siguientes, lo único que hay que hacer es ejecutarlo. Cuando haces uso de un compilador, este tiene acceso a todo el código de antemano, por ello que las optimizaciones que pueda realizar puedan ser mucho mayores que haciendo uso de un intérprete. Otra gran desventaja, es la dificultad de paralelizar el código sobre diferentes procesadores u optimizarlo sobre lenguaje vectorial haciendo uso de *GPU* (Unidad de Procesamiento Gráfico), algo que es necesario en el procesamiento de imágenes.

Para solventar este problema en *Python*, se han creado compiladores *JIT* (*Just-In-Time*), que permiten crear optimizaciones en tiempo de ejecución, haciendo uso de lectura por bloques y detección de estructuras como pueden ser bucles. En el caso de *Python*, destaca la librería Numba<sup>1</sup>, la cual es capaz de integrar este compilador *JIT* de una forma muy sencilla haciendo uso de decoradores, permitir optimizaciones y paralelización sobre múltiples threads, permitir paso de código a operaciones vectoriales *SIMD* (*Single Instruction and Multiple Data*) y poder hacer uso de *GPU* mediante la tecnología *CUDA* (*Compute Unified Device Architecture*) de *NVIDIA* y *ROCm* (*Radeon Open Compute*) de *AMD*.

### 5.1.2. Entorno de desarrollo

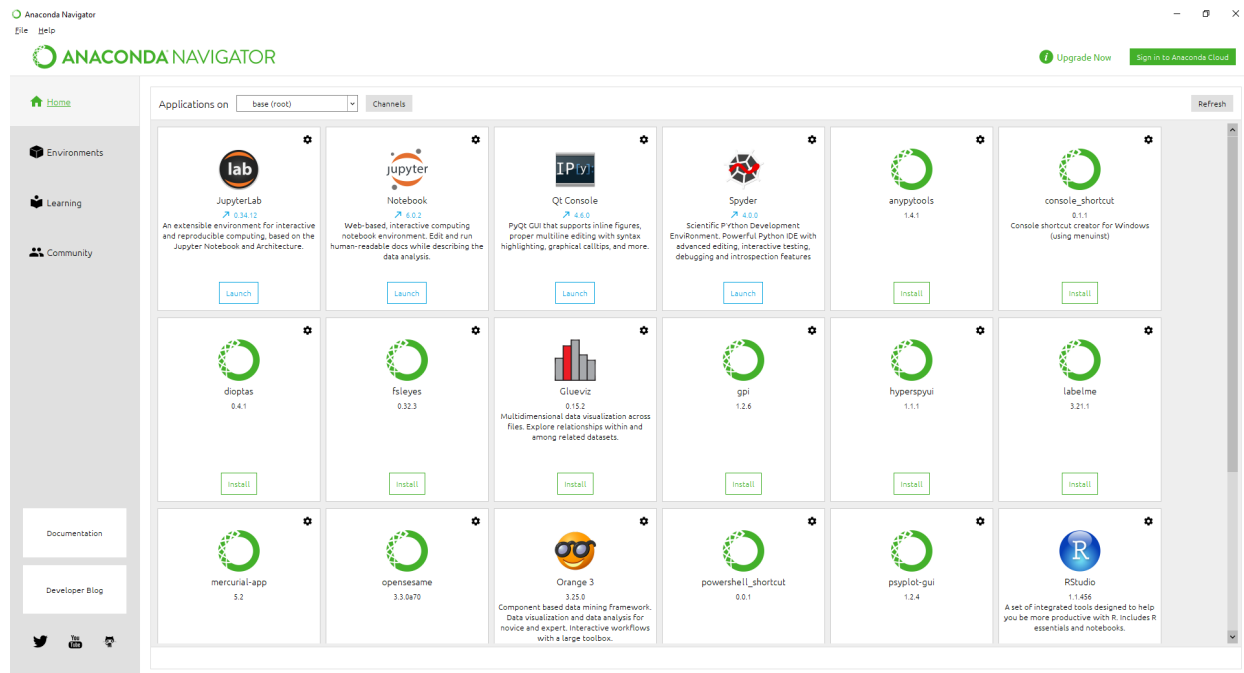
Un **entorno de desarrollo integrado** (comúnmente conocido como *IDE*), es un programa o plataforma que brinda al usuario de las herramientas de desarrollo necesarias para codificar las instrucciones del programa para su posterior ejecución. Entre esas herramientas se suele encontrar un editor de código, un depurador, un controlador de versiones, un enlazador a plataformas (como puede ser *GitHub* y *Amazon Web Services*) entre otros. En informática, existen a su vez distribuidores de entornos y paquetes (conocidos comúnmente como **suites**) capaces de brindar todo lo necesario para emprender un proyecto de código

---

<sup>1</sup><http://numba.pydata.org/>



de forma sencilla. Anaconda<sup>2</sup> es la suite referente al hacer uso de *Python* y *R*, conteniendo cerca de 20 millones de usuarios, 2.4 billones de paquetes y grandes entornos de desarrollo como puede ser *Spyder*, *Notebook Jupyter*, *Qt Console* y *RStudio* entre otros muchos tal como puede observarse en la Figura 5.1.



**Figura 5.1:** Sistema de entornos de desarrollo en Anaconda.

En la aplicación de la suite, además de encontrar numerosas guías y enlaces a su comunidad de usuarios, se puede encontrar también un fácil sistema de descarga de paquetes mediante su gestor *Conda*. Este sistema hace uso de un sistema de directorios para el almacenamiento de paquetes usados en diferentes proyectos (conocidos como Conda environments). Cada environment o entorno se puede enlazar y hacer uso desde otros *IDE* como puede ser *PyCharm*, *PyDev*, etc. Al enlazarlo, se creará una carpeta desde donde se podrá acceder a una terminal, permitiendo hacer uso de Conda para la instalación de numerosos paquetes (aunque existen variaciones para la descarga como *Pip*). Por lo tanto, aquellos paquetes encontrados en un entorno no pueden hacerse uso desde otro, brindando la característica de

<sup>2</sup><https://www.anaconda.com/>

encapsulamiento al proyecto (ver Figura 5.2).

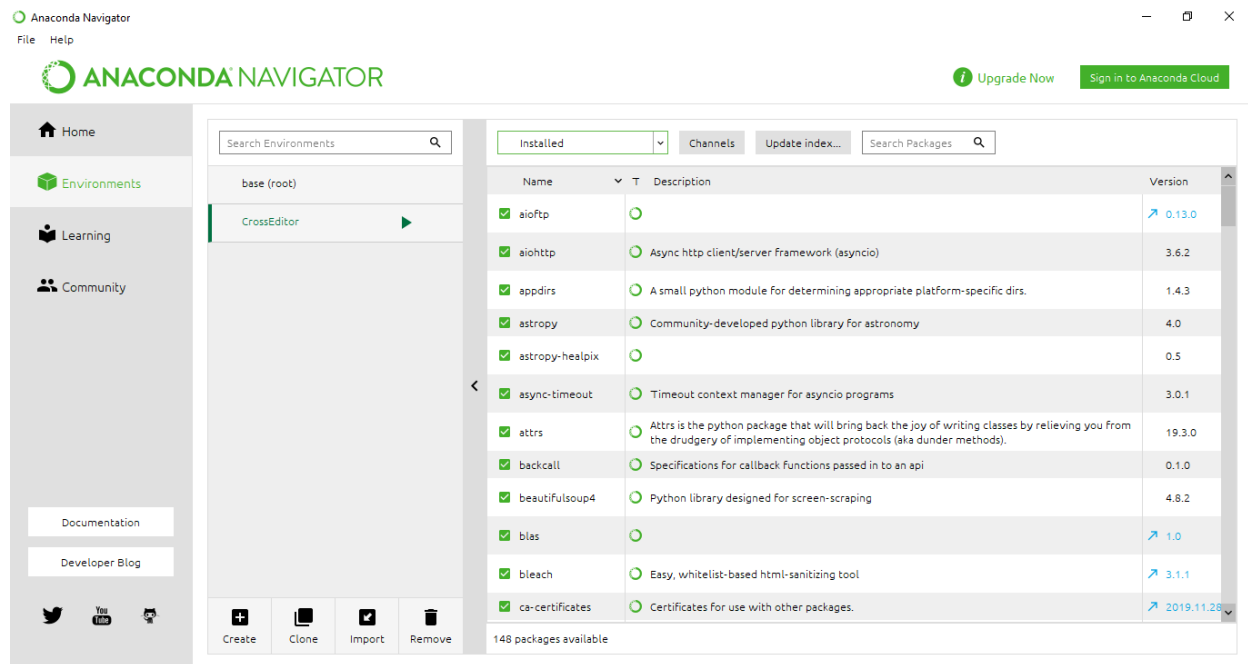


Figura 5.2: Sistema de paquetes en Anaconda.

En el proyecto, usaremos como suite Anaconda y como entorno de desarrollo *PyCharm*<sup>3</sup>. *PyCharm* es un *IDE* desarrollado por la empresa *JetBrains*, que brindará un *IDE* potente y sencillo de usar (ver Figura 5.3). Tal y como puede observarse, en el lado izquierdo se encontrará toda la jerarquía de carpetas y módulos, en su lado derecho, el código de un archivo o archivos (estructurado en pestañas), en la parte superior todas las herramientas (como puede ser la vista, las opciones, las opciones del intérprete) y en la parte inferior, todo el panel de salida de datos junto las opciones del depurador de código.

Otra gran característica que posee *PyCharm* es incluir en la interfaz las opciones del repositorio *GitHub* al que se subirá el proyecto, desde la que podemos acceder a las funciones de descarga o actualización de este.

<sup>3</sup><https://www.jetbrains.com/es-es/pycharm/>

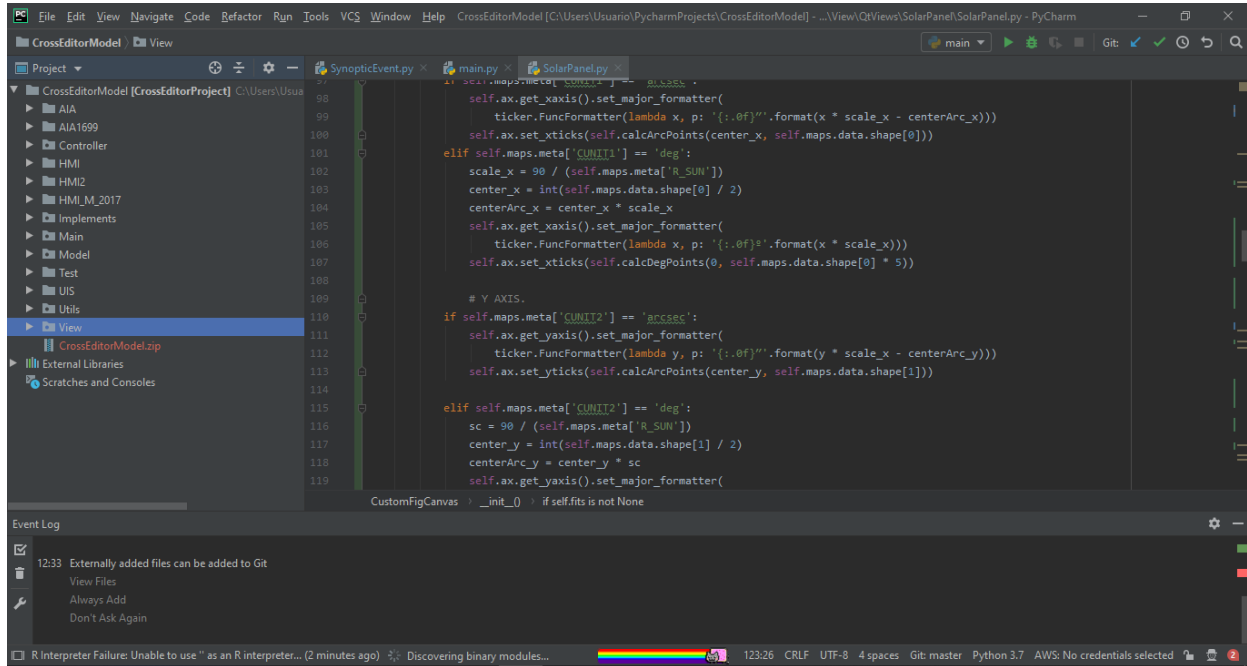


Figura 5.3: Interfaz gráfica de PyCharm.

### 5.1.3. Repositorio

*GitHub* es una plataforma orientada a desarrolladores cuyas principales funcionalidades son la gestión, almacenamiento, planificación, control de versiones y su comunidad entre otras, permitiendo en un futuro poder crear nuevas colaboraciones con otros desarrolladores y usuarios.

Debido a su gran sencillez y utilidad, el proyecto creado será almacenado en un repositorio público en la plataforma de software libre *GitHub* bajo el nombre *CrossEditorProject*<sup>4</sup>.

## 5.2. Gestión del proyecto

Para gestionar el proyecto de programación cuya finalidad es lograr el mapa sinóptico ideado, se estudiará qué paquetes harán falta para realizar todo el proceso. Una vez examinados los diferentes módulos que puedan aplicarse, hay que crear la estructura principal o esqueleto de la aplicación. Para ello, se tratará de cómo estructurar un sistema de comandos

<sup>4</sup><https://github.com/gabgarar/CrossEditorProject>

y qué patrones se les dará uso para una sencilla aplicabilidad y escalabilidad de la aplicación en un futuro.

### 5.2.1. Fase de investigación sobre paquetes

Entre los paquetes que serán aplicados a lo largo del proyecto destacan:

- **NumPy**: paquete básico y potente, usado para cálculo científico en *Python* fundamentado sobre una licencia *BSD-3-Clause* y mantenido por su propia comunidad sobre un repositorio de *GitHub*. *NumPy*<sup>5</sup> ofrece una gran variedad de tipologías, variables y funciones, que pueden ser aplicadas e integradas en distintos paquetes. De ella se hará uso: sus variables y tipos, las matrices e hiper-matrices y distintas funciones de reconversión entre tipos entre muchas más.
- **Matplotlib**: es el paquete por referencia para realizar gráficos sobre infinidad de datos almacenados en listas, colecciones, etc. Todo el cálculo usado se realiza sobre *NumPy*, por ello que esta librería sea tratada como una extensión de esta. Al igual que *NumPy*, *Matplotlib* está mantenida por una gran comunidad bajo una licencia *PSF*. *Matplotlib*<sup>6</sup> junto a *Numpy* se considera un gran competidor de *Matlab*, por ello, que la forma de realizar sus gráficos sea similar. A su vez, *Matplotlib* permite incluir sus gráficos en distintas plataformas destinadas a interfaces de usuario como puede ser *Qt*, *Tkinter* y *Kivy* entre otras.
- **Numba**: será el paquete que permita hacer uso de compilador *JIT* introducido anteriormente. Este compilador es útil a la hora de querer optimizar operaciones sobre tipos y estructuras de la librería *NumPy*, ya que está optimizada para ello. Numba puede aplicarse sobre una amplia gama de arquitecturas de 64-bits, permitiendo un uso muy sencillo de la misma mediante decoradores (llamadas implícitas de *Python* mediante `@jit`).

---

<sup>5</sup><https://numpy.org/>

<sup>6</sup><https://matplotlib.org/>

- **SunPy**: el paquete del que haremos uso para descargar los ficheros, poder tratarlos y almacenarlos mientras se haga uso de la aplicación. Su comunidad de desarrolladores tienen la principal finalidad de desarrollar y proveer de código para la comunidad científica cuya aplicabilidad sea en el campo de la física solar. SunPy<sup>7</sup> brinda de un gran número de herramientas para poder tratar desde imágenes, proyecciones, serie de datos y coordenadas entre muchas otras.
- **PyQt**: *Qt* es un *framework* orientado a al desarrollo de interfaces gráficas sobre C++. *Python* puede hacer uso de dicha plataforma mediante técnicas de *binding*, permitiendo hacer uso de las librerías de Qt<sup>8</sup> escritas sobre lenguaje C++. *PyQt* está basada en numerosas licencias, desde libres bajo la licencia *GPL* o comerciales para su puesta en mercado.

Estas librerías serán los pilares de la aplicación, usando como base las propias imágenes que serán almacenadas mediante matrices de *NumPy* en objetos de la librería *Sunpy*. Para el tratamiento de las imágenes, se hará uso de los métodos y variables que brindará *NumPy*, optimizando dichos métodos con funcionalidad de la librería *Numba*. Para la muestra de las imágenes y estructuras se utilizarán los distintos métodos brindados por la librería *Matplotlib* y para la interfaz de usuario, será desarrollada bajo los elementos que nos brinda *Qt*.

### 5.2.2. Estructuración del proyecto

En un proyecto de programación existen diversas formas de abarcar un mismo problema. Para su estructuración, el primer paso recomendado es la elección del patrón o patrones de los que hará uso el proyecto, debido a que actuarán dichos patrones como pilar de paso de información entre las diferentes partes.

Las aplicaciones que hagan uso de una interfaz de usuario (o varias), suelen estar desarrolladas bajo un **patrón modelo-vista-controlador** (reconocido como patrón *MVC*) debido

---

<sup>7</sup><https://sunpy.org/project/>

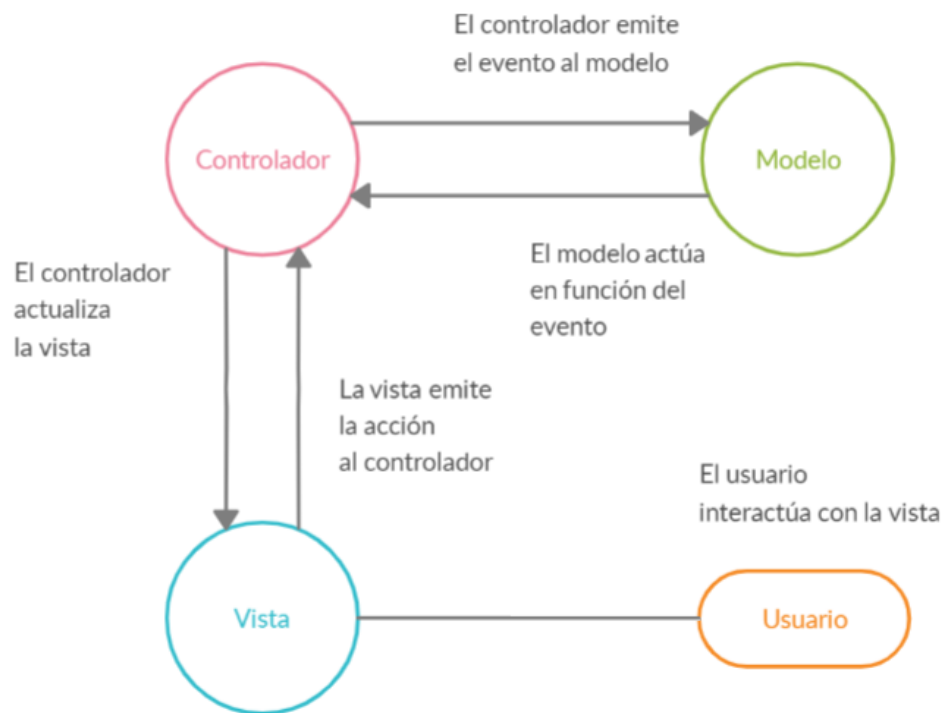
<sup>8</sup><https://www.qt.io/>

a su facilidad de uso y su escalado, por ello que se use como pilar en este proyecto (ver Figura 5.4). En un patrón *MVC*:

- **Modelo:** Será quién se encargue de almacenar y tratar la información a la hora de querer hacer alguna consulta o tratamiento sobre los datos. Actúa como agente de los datos, por lo que usualmente se le ofrecerá una consulta en forma de evento y, en función de este, actuará de una forma u otra sobre el sistema contenedor de las distintas imágenes.
- **Vista:** será la encargada de ofrecer una interfaz gráfica al usuario con la cual podrá interactuar. La vista en la aplicación será una terminal de uso o la propia interfaz de usuario realizada sobre *Qt*.
- **Controlador:** será el intermediario entre la vista y el modelo, permitiendo modificar el paso de información para adecuarlo a cada parte.

El modelo, dada su funcionalidad, podrá dividirse en dos partes: la parte de *almacenamiento de los datos* y la de *operaciones*. La de almacenamiento puede hacer uso de bases de datos o archivos en uno o más servidores (en el caso del proyecto, los archivos son descargados haciendo uso de un evento y almacenados de forma local). La de operaciones, se basará en una jerarquía de eventos estructurada en forma de árbol. Dicho árbol estará compuesto por distintos módulos y submódulos. Para esta jerarquía se hará uso del **patrón parser** conocido también como patrón de intérprete.

Una vez el modelo ejecuta el evento asociado al comando y se tratan los datos por parte del gestor del modelo, la vista en ciertas ocasiones deberá ser actualizada, debido a que contiene elementos como gráficos y tablas cuyos valores han sido modificados. Por ello que el controlador no solo enviará peticiones al modelo, sino que a su vez también se encargará de notificar los cambios producidos por los eventos a la propia interfaz, para que actualice sus estructuras que saca a la vista al usuario. Para ello, sobre el modelo vista controlador, se le añadirá el **patrón observador**. Este patrón permitirá a los distintos observadores del



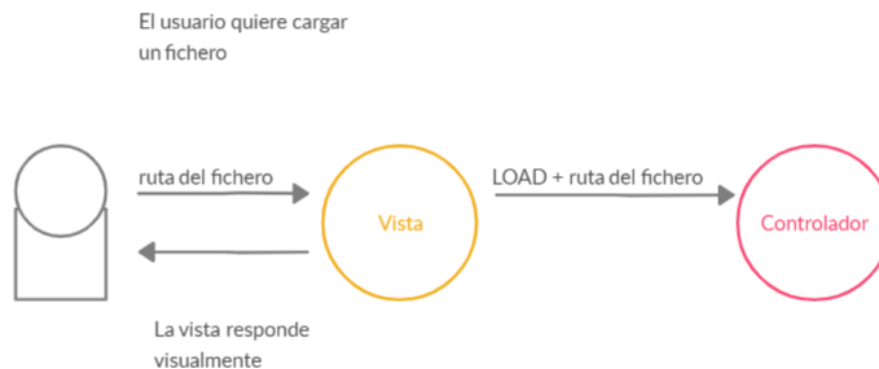
**Figura 5.4:** *Diagrama patrón MVC.*

sistema contenedor, ser actualizados y notificados en función de los eventos ejecutados sobre el sistema.

### 5.2.3. Sistema de la información del modelo

El sistema de información comenzará por una orden o comando realizado desde una terminal o la propia interfaz al interactuar el usuario con la misma. Esta interfaz será denominada **vista** tal y como se ha definido anteriormente. Por ello que la primera fase en el paso de información dentro de la aplicación será la interacción usuario-vista.

Dicha interacción provocará que la vista notifique al controlador con una secuencia de información. Dicha secuencia estará compuesta por el identificador del evento y los parámetros necesarios para realizar la consulta sobre el modelo. Si por ejemplo el usuario quiere abrir un fichero, la vista le requerirá una ruta del fichero al usuario por la interfaz, y tras obtenerla, la vista generará el comando o paso de información y será enviado al controlador (ver Figura 5.5).



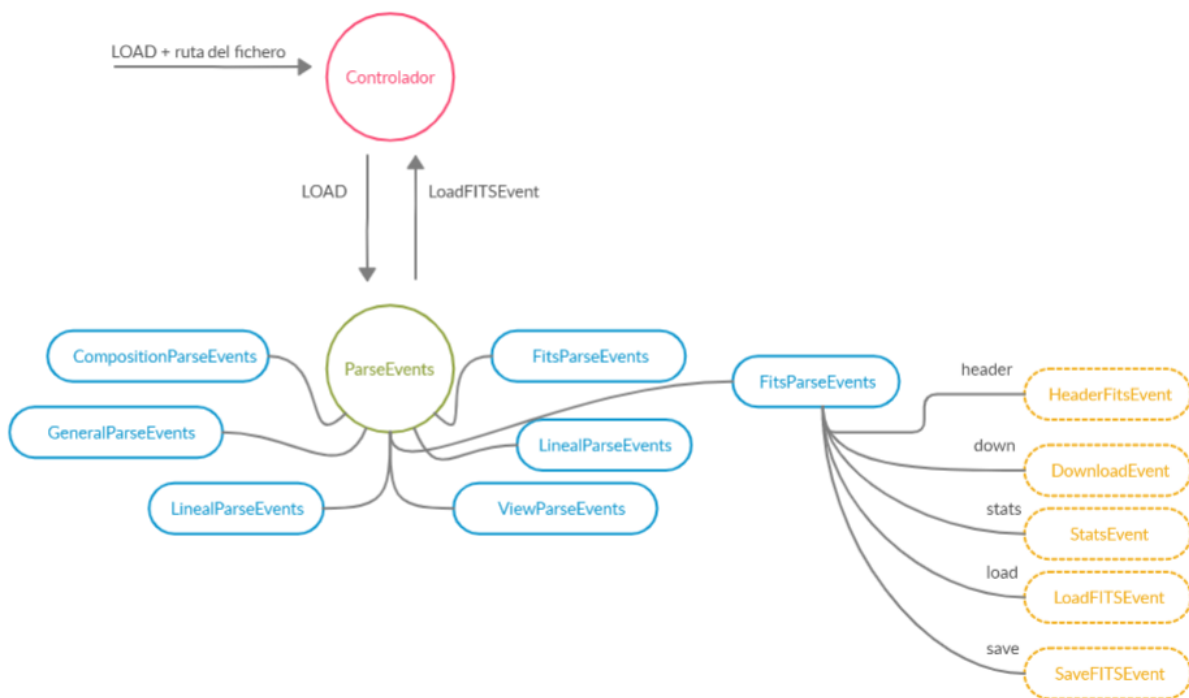
**Figura 5.5:** Paso de información entre la vista y el controlador.

Una vez tenga el controlador el comando, lo que hará será formatearlo generando una lista cuyo primer elemento será el identificador del evento y los siguientes, los parámetros de la consulta. Tras ello, el controlador hará uso del objeto **parser** para obtener el evento



ligado al identificador.

Tal y como puede observarse en la Figura 5.6, en la aplicación se harán uso de dos tipos de parser: el primero será un parser para eventos, el cual tendrá una lista de objetos evento que cuelguen de dicho parser (por ejemplo: el *FitsParseEvents* identificado en azul). El segundo será un parser más general, que actuará como nodo padre de los distintos parser para eventos, el cual permitirá una mayor modularización (en el diagrama será *ParseEvents*, identificado en color verde).



**Figura 5.6:** Paso de información entre el *parserEvents* y el controlador.

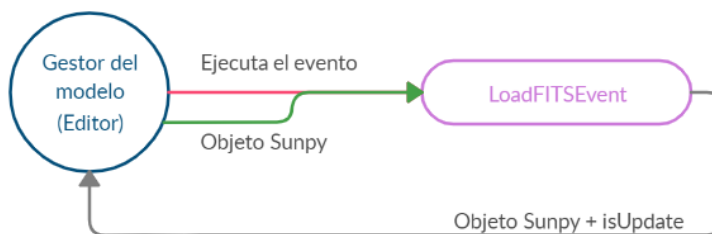
Una vez tenga el controlador el objeto evento adecuado al comando, se le pasa los parámetros junto el objeto evento al modelo. Dentro del modelo, el sistema que contiene el conjunto de imágenes junto a toda su información relativa ligada será el **sistema contenedor** mediante un objeto Sunpy (identificado en azul oscuro en la Figura 5.7).

Este sistema no es capaz de ejecutar ningún evento, solo puede acceder a su contenido

mediante distintos métodos, por lo que tendrá que existir un gestor que se encargue de dichas ejecuciones. Dicho gestor será el **gestor del modelo** el cual será capaz de ejecutar los distintos eventos en orden de llegada (ver Figura 5.8).



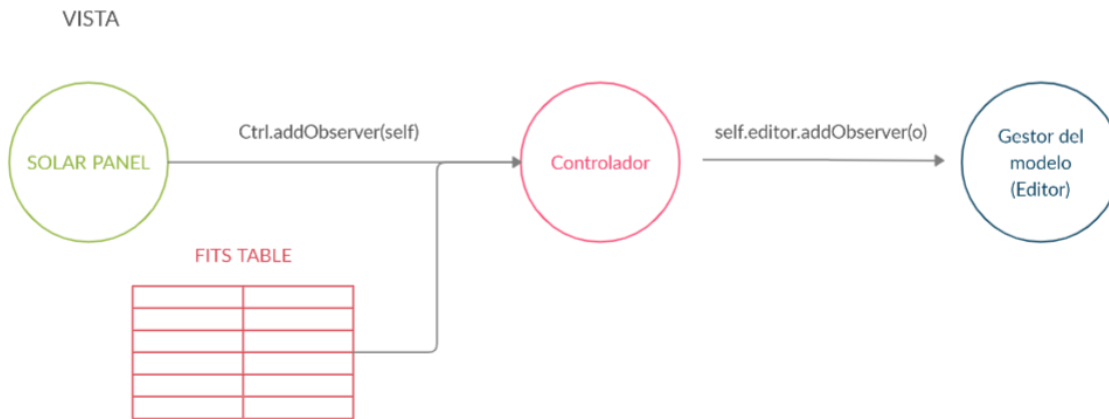
**Figura 5.7:** *Solicitud del objeto SunPy al sistema contenedor.*



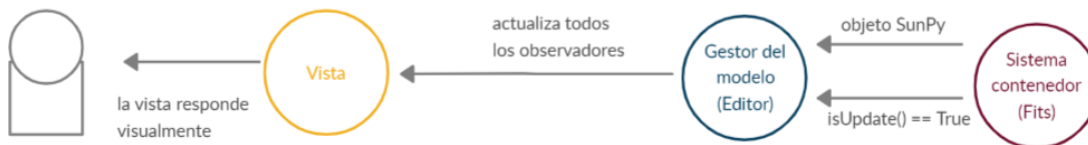
**Figura 5.8:** *Ejecución de un comando por parte del gestor del modelo.*

Una vez se ejecute el evento, el sistema contenedor se actualizará. Para controlar si un evento ha realizado una modificación sobre el sistema contenedor o simplemente consultaba una serie de valores, se hará uso de un parámetro para controlarlo. Esto permitirá saber cuándo es necesario actualizar los diferentes observadores del sistema contenedor. Como quién gestiona todas las operaciones sobre la imagen es el gestor, el mismo contendrá una lista de todas las estructuras observadoras del sistema.

Para que una estructura como una tabla o un frame en la vista se considere observador, se deberá de indicar al controlador dicha acción, y posteriormente él mismo, le pasará la información al gestor del modelo para incluir dichas estructuras como observadores. Dicho proceso se puede observar en la Figura 5.9.



**Figura 5.9:** *Declaración de una estructura como observador.*



**Figura 5.10:** *Actualización de los observadores tras una modificación sobre el sistema contenedor.*

Los observadores se añaden nada más iniciar la aplicación para que una vez se ejecute un evento, siempre que se modifique el sistema contenedor, se le pase la información a cada observador y estos puedan actualizarse debidamente (ver Figura 5.10).

Una vez se ha estructurado todo el paso de información y las partes necesarias para que la aplicación funcione, el siguiente paso es incluir la propia funcionalidad al proyecto. La funcionalidad no depende únicamente de los eventos sino también de las distintas opciones que permita la vista. La funcionalidad dada por los eventos, en resumen, se dividirá en una serie de módulos, en el que cada uno contendrá una serie de eventos agrupados por temática y un parser que permita acceder a los mismos. Toda la funcionalidad principal será tratada en el siguiente capítulo.



# Capítulo 6

## Funcionalidad de la aplicación

La funcionalidad de la aplicación viene fundamentada por la variedad de eventos que se permite realizar sobre el sistema contenedor, por las opciones de visualización y la forma en la que puede interactuar el usuario con la misma. Dada la estructura jerárquica de comandos y eventos aplicada en el desarrollo de la aplicación, los eventos serán agrupados por funcionalidad en distintos módulos o carpetas. Debido a que la funcionalidad que puede introducirse a una aplicación de edición de imagen puede ser descomunal, la funcionalidad será explicada en dos grupos, la funcionalidad principal y la secundaria.

### 6.1. Agrupamiento de eventos

Cada carpeta o módulo contendrá una serie de eventos y un *parser* que les permita acceder. Dichos módulos son los siguientes:

- **Eventos FITS:** todos aquellos eventos encargados de descargar, cargar, guardar y consultar sobre el sistema contenedor.
- **Eventos generales:** aquellos que actúan sobre el flujo de la aplicación como puede ser cerrarla, reiniciar, volver hacia atrás o hacia delante en la edición de la imagen.
- **Eventos lineales:** contienen diversos filtros aplicables al conjunto de imágenes encontradas en el contenedor.

- **Eventos convolucionales:** permite aplicar distintas convoluciones sobre el conjunto de imágenes.
- **Eventos de formato:** permite cambiar el formato de imagen, normalizarla en unos rangos, cambiar el tipo de variable que define una imagen entre otros.
- **Eventos de composición:** agrupa transformaciones sobre la imagen como puede ser la polar o las distintas enseñadas en el capítulo anterior. A su vez contiene el evento para componer los mapas sinópticos.
- **Eventos de vista:** permite enseñar la imagen o una secuencia en orden cronológico en el caso en el que se agrupe más de una imagen. Además permite mostrar histograma, gradiente del pseudocolor o el diagrama de *Fourier* entre otros.

## 6.2. Funcionalidades principales

Una vez se ha visto cómo se agrupan el sistema jerárquico de eventos, se tratará de explicar cuáles son las funcionales básicas y principales que ofrece la aplicación entre todas. De entre ellas podemos destacar las siguientes:

### 6.2.1. Servidores de consulta de los instrumentos

Desde el paquete *SunPy* se puede hacer uso el objeto *Fido*, el cual permitirá consultar información sobre numerosos servidores desde donde se puede obtener diversos ficheros hasta serie de datos actualizados cada un pequeño periodo de tiempo.

Su aplicación es sencilla, necesita como parámetros el intervalo de tiempo entre dos fechas, el instrumento e información específica de dicho instrumento (como puede ser de la sonda *SDO*, el instrumento *AIA* y una longitud de onda de 1600 Ångstrom). A su vez, puede especificarse a la consulta cada cuanto tiempo puede descargarse o informarse sobre una imagen.

Dado este procedimiento, con un único comando se realiza una consulta y una posterior descarga de dichos ficheros (en el caso de la sonda SDO, las imágenes pueden visualizarse a tiempo real y poder ser descargadas en función de una ficha desde la página principal de la sonda<sup>1</sup>). Un ejemplo del comando que describe el evento será el siguiente:

**DOWN 2016/01/01 2016/02/01 aia 171 1440**

Dicho comando define la consulta y la descarga de todos los ficheros referentes al instrumento *SDO/AIA* con una longitud de onda de 171 [Å], comprendidos entre el 1 de enero de 2016 al 1 de febrero de 2016 y cada toma de imagen en un intervalo de tiempo de 24 horas.

La consulta dará como resultado la Tabla 6.1 indicando en el ejemplo que se han encontrado 31 ficheros que entran dentro del criterio de búsqueda, haciendo uso del cliente que contiene los distintos ficheros *FITS*. Una vez se ha realizado la consulta, se descargarán los ficheros correspondientes a la consulta.

Start Time [1]	End Time [1]	Source	...	Type	Wavelength [2]
2016-01-22 00:00:10	2016-01-22 00:00:11	SDO	...	FULLDISK	171.0 .. 171.0
2016-01-25 00:00:10	2016-01-25 00:00:11	SDO	...	FULLDISK	171.0 .. 171.0
2016-01-07 00:00:10	2016-01-07 00:00:11	SDO	...	FULLDISK	171.0 .. 171.0
2016-01-24 00:00:10	2016-01-24 00:00:11	SDO	...	FULLDISK	171.0 .. 171.0
2016-01-01 00:00:10	2016-01-01 00:00:11	SDO	...	FULLDISK	171.0 .. 171.0
2016-01-04 00:00:10	2016-01-04 00:00:11	SDO	...	FULLDISK	171.0 .. 171.0
...	...	...	...	...	...
2016-01-19 00:00:10	2016-01-19 00:00:11	SDO	...	FULLDISK	171.0 .. 171.0
2016-01-13 00:00:10	2016-01-13 00:00:11	SDO	...	FULLDISK	171.0 .. 171.0
2016-01-11 00:00:10	2016-01-11 00:00:11	SDO	...	FULLDISK	171.0 .. 171.0
2016-01-23 00:00:10	2016-01-23 00:00:11	SDO	...	FULLDISK	171.0 .. 171.0
2016-01-03 00:00:10	2016-01-03 00:00:11	SDO	...	FULLDISK	171.0 .. 171.0

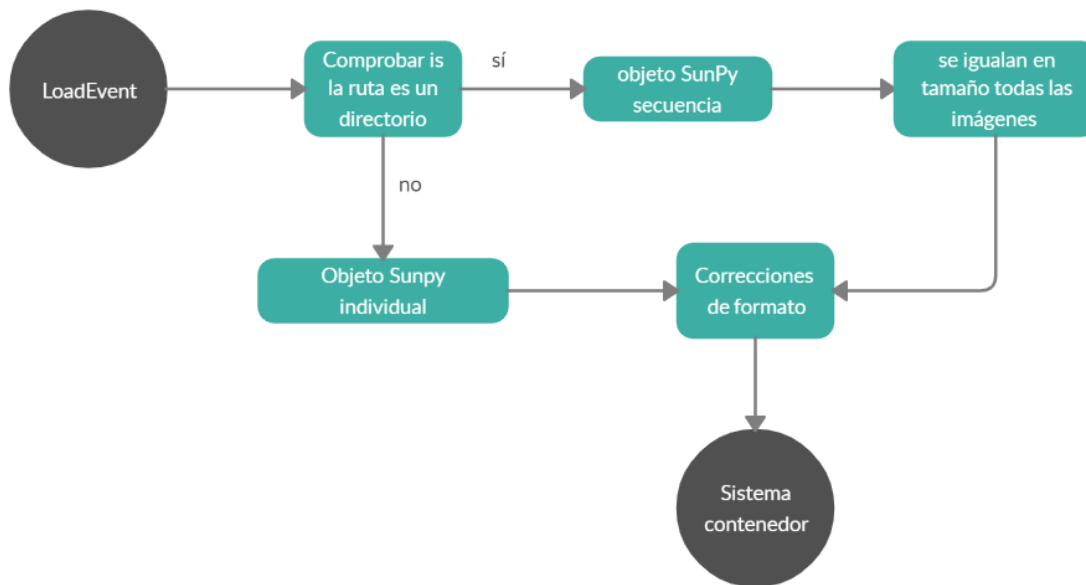
**Tabla 6.1:** Consulta sobre el cliente que contiene los ficheros de la sonda SDO.

---

<sup>1</sup><https://sdo.gsfc.nasa.gov/>

### 6.2.2. Cargar una imagen o secuencia a la aplicación

Una vez se tengan los ficheros descargados en formato *FITS*, el siguiente paso será poder hacer uso de ellos desde la aplicación. Por ello se explicará el proceso de apertura y carga de un archivo de forma más exhaustiva (el cual puede verse en la Figura 6.1).



**Figura 6.1:** Cargar imágenes en un objeto *FITS*.

Como el proceso de paso por parámetro se ha introducido anteriormente y es similar para todos los eventos, esta parte será obviada. Una vez el gestor del modelo tiene el evento, se le pasa al evento el sistema contenedor junto a los parámetros para poder ser ejecutado por el gestor del modelo.

Al ejecutarlo, como el objeto *SunPy* permite cargar una o más imágenes, cada una con su propia cabecera *FITS*, el paso de cargar una secuencia o una sola imagen será similar. Cuando se tiene una secuencia, hay que igualar en tamaño todas las imágenes para poder hacer uso más tarde de la composición. Para ello, *SunPy* proyecta todas las imágenes y



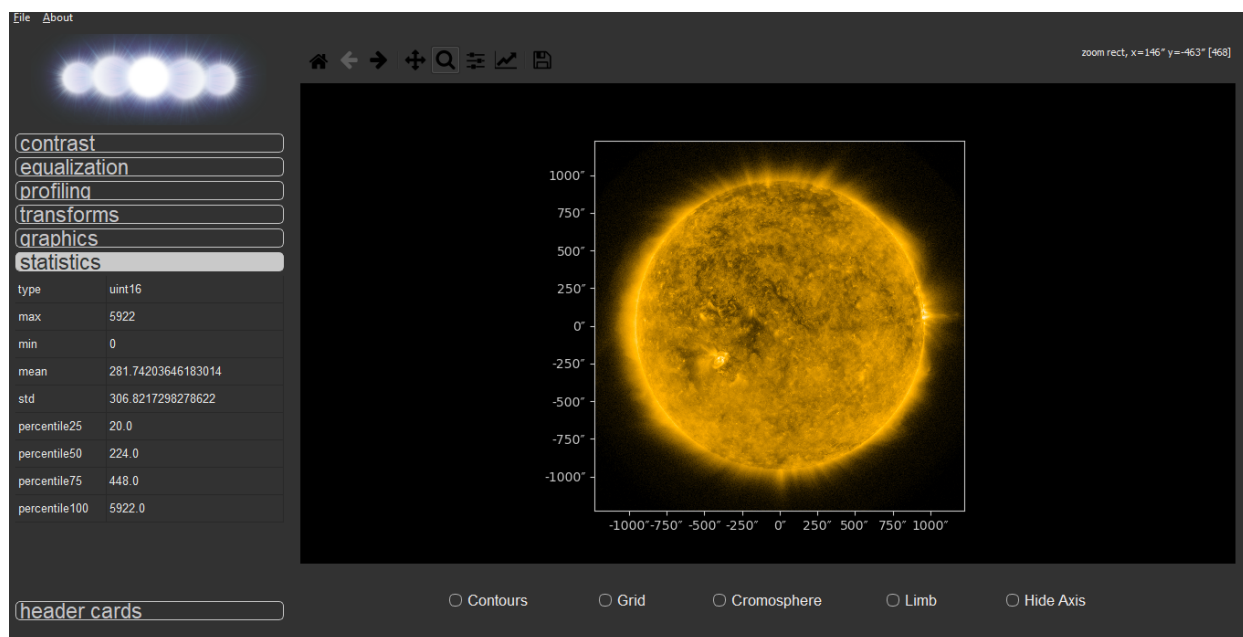
recorta en función del tamaño de la imagen más pequeña. Una vez hecho este proceso, se le añadirá el objeto *SunPy* al sistema contenedor.

Dentro del sistema contenedor, antes de añadir el objeto *SunPy*, este realizará unas correcciones necesarias para poder hacer un uso correcto del conjunto de imágenes. Entre esas correcciones estará la de realizar un casting sobre el propio formato del conjunto de imágenes. Es necesario debido a que ciertas funciones no permiten ciertos formatos de entrada o en el caso en el que sí lo permitan, el formato de salida tras ejecutar un evento puede variar respecto al inicial. En la aplicación se trabajará en todo momento con un tipo de datos de entero sin signo de 16 bits aunque para versiones futuras este formato variará. Otras correcciones necesarias son por ejemplo que cierto tipo de imágenes como las del instrumento *HMI* es necesario normalizarlas para poder hacer un correcto uso de su pseudocolor. Esta corrección se realizará sobre cada imagen encontrada en el objeto *SunPy*.

Tras cargar una imagen, puede observarse cómo queda sobre la interfaz de usuario (ver Figura 6.2). En el caso en el que sea cargada una secuencia, se mostrará el paso de una imagen a otra sobre la interfaz.

### 6.2.3. Aplicar un filtro sobre un conjunto de imágenes

La aplicación de un filtro o una convolución se realiza cómo una transformación sobre la imagen. Dependiendo de las características del filtro, este puede hacer uso de convoluciones, filtros por píxel, lineales o por edición de frecuencias entre muchos otros. En el caso que se quiera aplicar una *corrección gamma* sobre la imagen, este filtro necesitará hacer uso de ciertos parámetros, principalmente uno denominado *factor gamma*. Dicho parámetro se le será pedido al usuario por la vista y haciendo uso de todo el proceso de paso de información, se llegará a aplicar el filtro correspondiente sobre el sistema contenedor haciendo uso de un evento. Una vez aplicado el evento, el sistema contenedor volverá a realizar las correcciones de formato necesarias para que no existan problemas a la hora de aplicar otros eventos sobre el propio sistema. El resultado de este sobre la imagen inicial con un *factor gamma* igual a



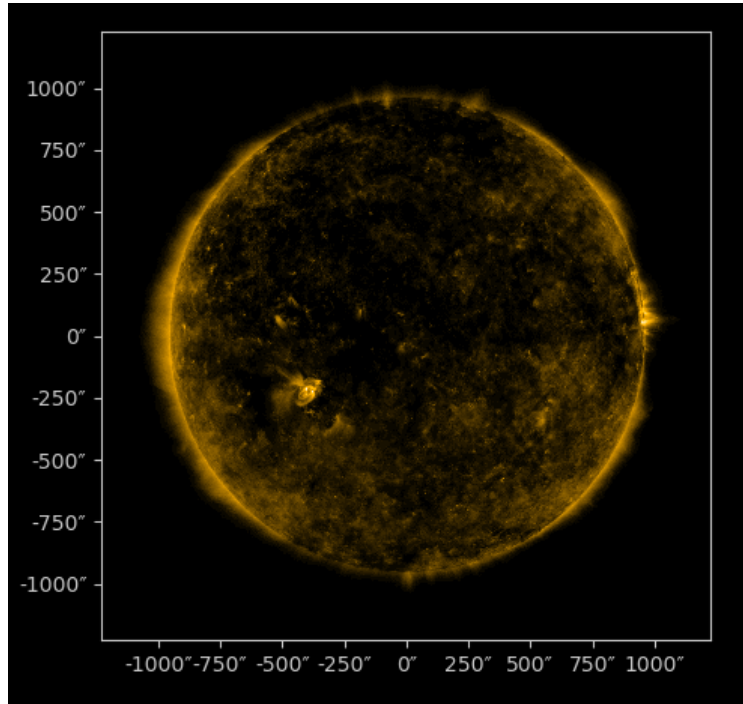
**Figura 6.2:** Imagen procedente del instrumento *SDO/AIA* 171 Ångstrom del día 17/08/2017 cargada en la interfaz.

2 podrá verse en la Figura 6.3.

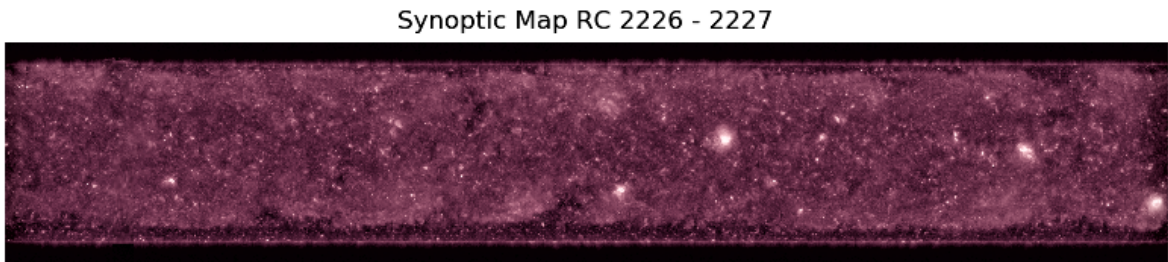
El resultado que puede observarse en este tipo de correcciones es una mayor oscuridad. En función del parámetro *gamma*, cuando es superior como en este caso a 1, las sombras toman mayor protagonismo, en cambio, cuando dicho factor es menor a 1, aumenta la luminosidad.

#### 6.2.4. Creación de un mapa sinóptico

Para que se pueda desarrollar un mapa sinóptico cuyo proceso fue descrito en el capítulo 4, la aplicación deberá haber cargado una serie de imágenes en el sistema contenedor previamente cómo una secuencia. Dicha secuencia deberá ser ordenada de forma cronológica, se deberá de nivelar el tamaño de todas sus imágenes y castear todas ellas a un tipo común. Para su realización se han tomado imágenes desde el día 6 de enero de 2020 hasta el 1 de marzo del mismo año, completando un total de dos rotaciones de Carrington, la 2226 y la 2227. Dichas imágenes proceden del instrumento *SDO/AIA* en una longitud de onda de 211 Ångstrom, dando como resultado el encontrado en la Figura 6.4.



**Figura 6.3:** *Aplicación de un ajuste gamma.*



**Figura 6.4:** *Composición de un mapa sinóptico a través de una secuencia de imágenes.*

Como se explica en el apartado histórico, estos mapas suelen hacerse para contener distintas rotaciones, permitiendo realizar distintas comparaciones con otras composiciones en distintos ciclos solares o épocas dentro del mismo.

## 6.3. Funcionalidades secundarias

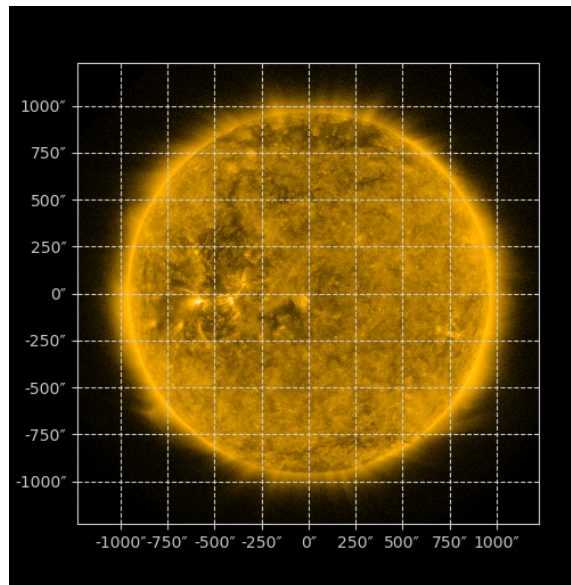
Son funcionalidades no obligatoriamente necesarias para el usuario, aunque se introducen en la aplicación para que pueda obtener una mayor experiencia al editar una o más imágenes solares o simplemente consultar ciertas partes. Dicha funcionalidad adicional permitirá realizar consultas en la propia imagen como elementos estadísticos o coordenadas entre otras opciones. De entre dichas funcionalidades destacan las siguientes:

### 6.3.1. Toppings

Se han añadido distintas funcionalidades que permitan superponer elementos sobre el *frame* de muestra, sin que modifiquen la estructura de la imagen pudiéndose aplicar varios de ellos a la vez. Estos elementos serán denominados *toppings*. De entre ellos destacan:

#### 1) Mallado sobre los ejes

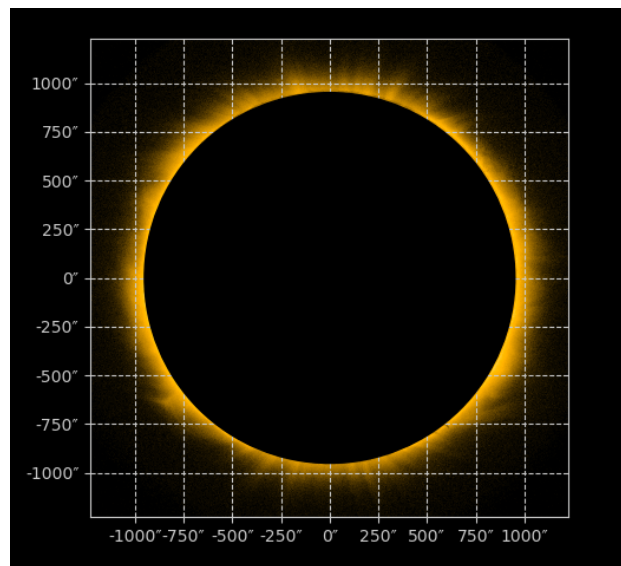
Permitirá superponer una malla o *grid* sobre los ejes del gráfico, permitiendo servir como apoyo o ayuda a la hora de situar un punto o coordenada sobre la imagen. Como resultado se obtiene el encontrado en la Figura 6.5.



**Figura 6.5:** *Aplicación de una malla sobre la imagen.*

## 2) Cromosfera

En el caso de la cromosfera, lo que hará será centrar un círculo de radio Sol sobre la imagen, permitiendo dejar visible las zonas exteriores al disco solar. En imágenes solares como las obtenidas por el instrumento *SDO/AIA* a 171 Ångstrom, permite ver parte de la cromosfera y la corona sobre la imagen. El resultado de aplicar este filtro es el visible en la Figura 6.6.



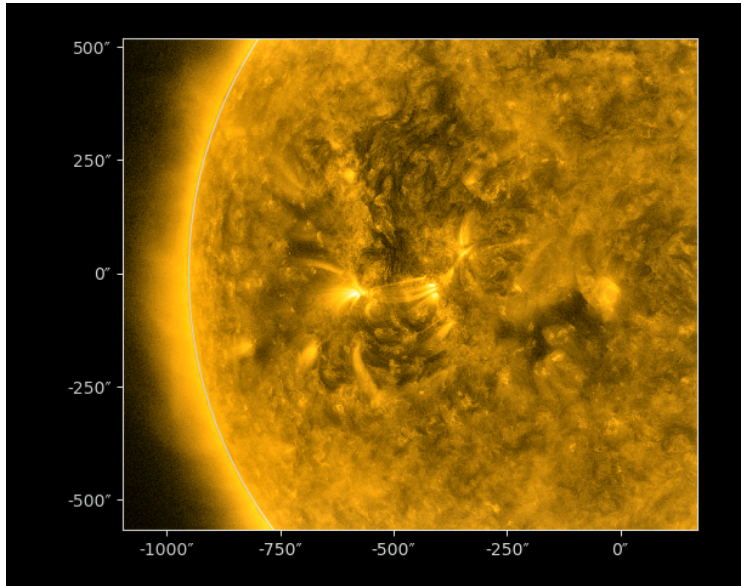
**Figura 6.6:** *Cromosfera en la imagen.*

## 3) Limbo

Se entenderá como limbo solar al borde del cuerpo observable y usualmente este dato es dado en segundos de arco sobre la cabecera *FITS*. El limbo solar permite saber dónde delimita la parte de superficie solar de la corona como ejemplo. Su resultado visible puede observarse en la Figura 6.7.

## 4) Contornos

Otra funcionalidad que permite la aplicación es rodear las distintas estructuras encontradas en la imagen siendo de utilidad para determinar qué puntos o zonas son más brillantes



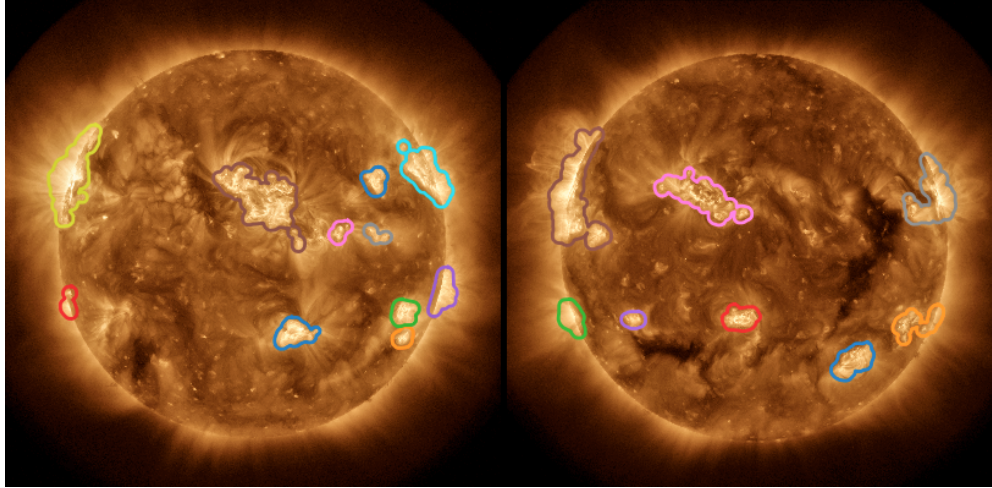
**Figura 6.7:** *Limbo sobre la imagen.*

que las demás, rodeándolas con un contorno sobre la imagen original. Para encontrarlas, usualmente se hace uso de distintos filtros. Se suele empezar con una binarización de la imagen por tramos, dejando como blancos o negros aquellas zonas que superen un valor. Una vez realizada esta operación, se aplica un filtro de realce de bordes como puede ser un filtro *canny* o *sobel* y una vez se tiene el borde, se superpone sobre la imagen original. En ciertos casos en los que la imagen tenga bastantes picos de intensidades, se aplica un selector de área del contorno mostrando únicamente aquellos contornos que superan un valor de área contenida. Un posible resultado de ello puede verse en la Figura 6.8, mostrando en dos imágenes distintas las distintas zonas luminosas.

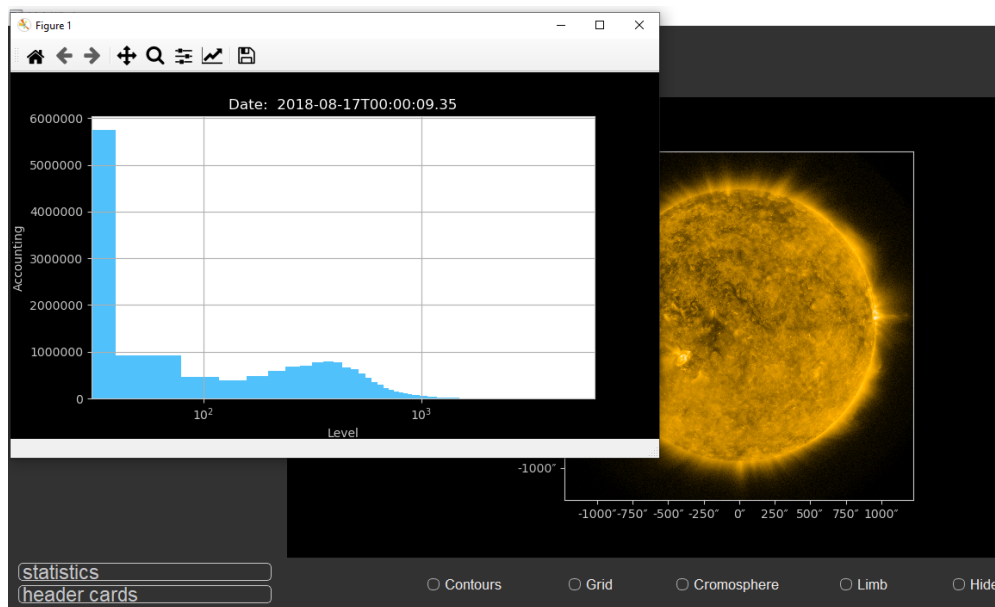
### 6.3.2. Histogramas

Las imágenes se muestran haciendo uso de píxeles. Cada píxel viene descrito con una tonalidad dada por un valor. Cuanto más oscuro sea un píxel, más bajo será su valor. Para generalizar las tonalidades y representar su distribución por la imagen, se hace uso de diagramas de frecuencias o histogramas. Una representación de la imagen principal sería la visible en la Figura 6.9.





**Figura 6.8:** *Contornos sobre imágenes solares.*



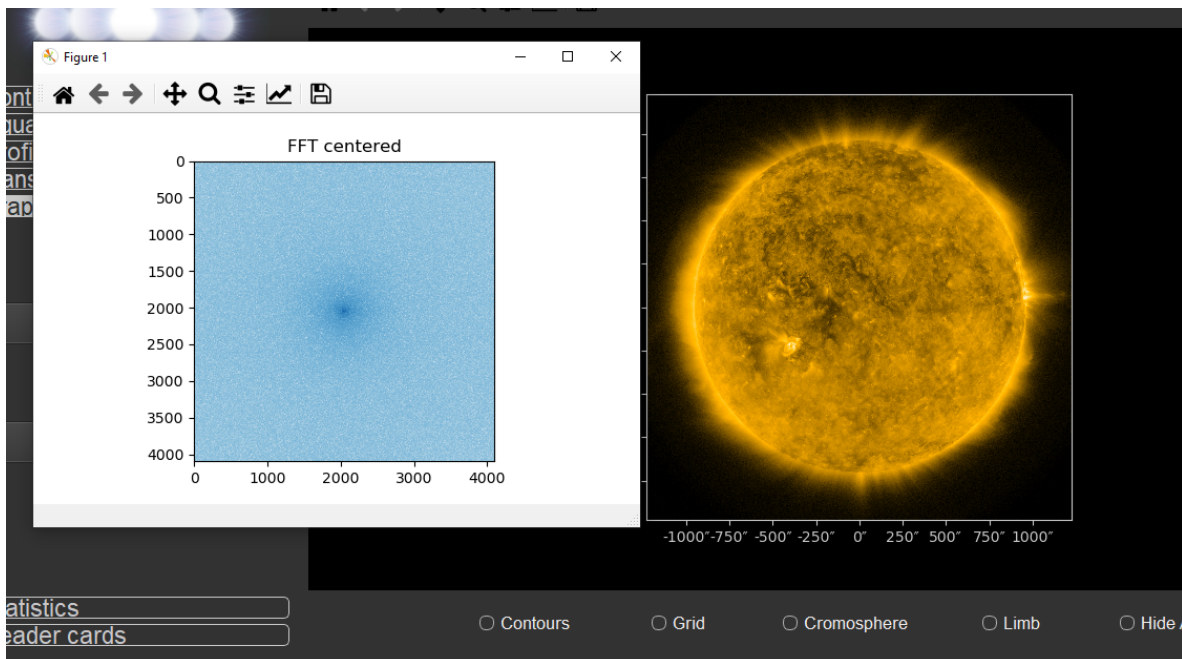
**Figura 6.9:** *Histograma de una imagen solar.*

Cuanto más oscura sea una imagen, más desplazado hacia la izquierda se mostrará el diagrama de frecuencias. En cambio, cuanto más luminosas, más hacia su derecha. Para su representación, en el eje de ordenadas se muestran cuantas veces aparece un valor y en el eje de abscisas, el propio valor asociado a una tonalidad.

### 6.3.3. Diagrama DFT

En teoría de señales, se hace uso de la transformada de *Fourier* para describir una serie de ondas fundamentales que en su composición describe la propia señal. Si se toma una imagen como sistema bidimensional, se puede aplicar la transformada de *Fourier* discreta para obtener el espectro de frecuencias. Este espectro es útil, debido a que se pueden realizar distintos filtros sobre el espectro de frecuencias para obtener distintos resultados. Los filtros aplicados en el espectro de frecuencias suelen ser tratados como filtros pasa frecuencia, permitiendo dejar únicamente ciertas bandas de frecuencia actuando como filtros suavizantes o filtro de realce entre otros. A su vez permite también determinar si una imagen está bajo la aparición de ruido como puede ser el de vibraciones del instrumento, permitiendo en muchas ocasiones, poder reducirlo o eliminarlo debido a que presentan patrones en el espectro de frecuencias.

Dicho espectro puede visualizarse en la Figura 6.10. En el espectro, se han centrado las bajas frecuencias al centro y las altas frecuencias en las zonas radiales.

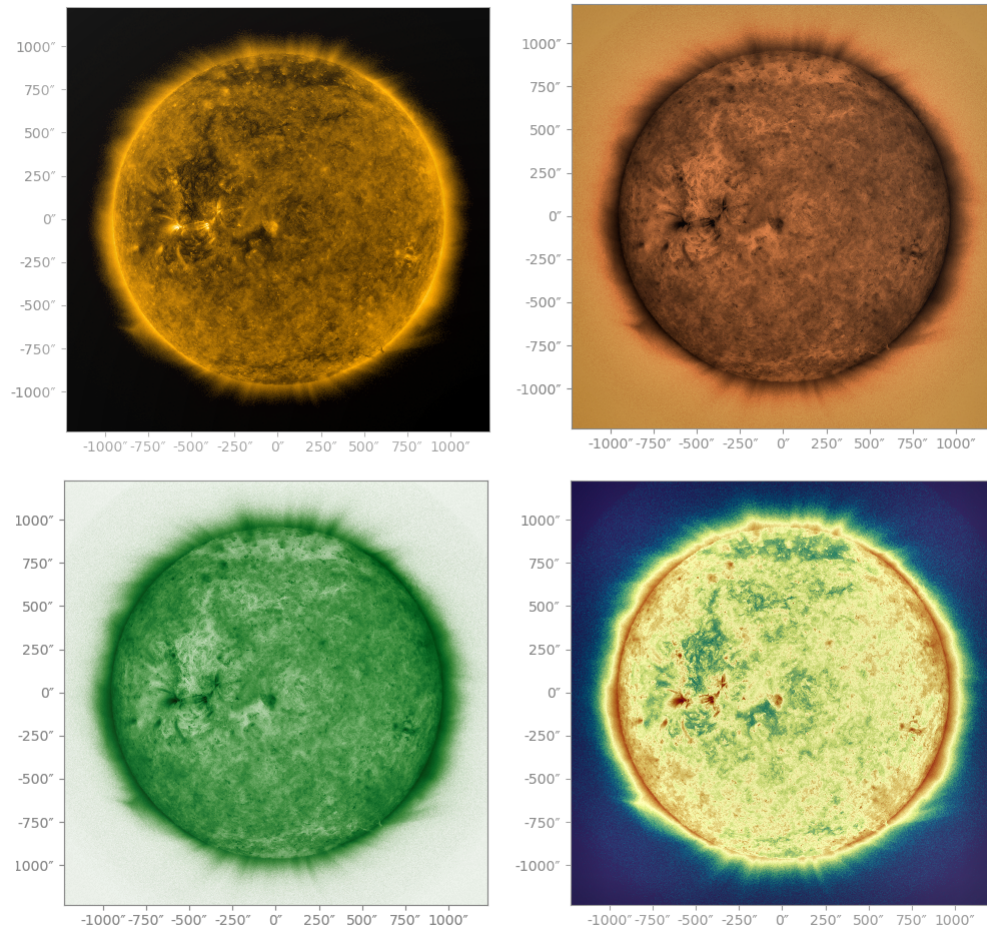


**Figura 6.10:** *Espectro de frecuencias de una imagen solar.*



### 6.3.4. Opciones de vista sobre la imagen

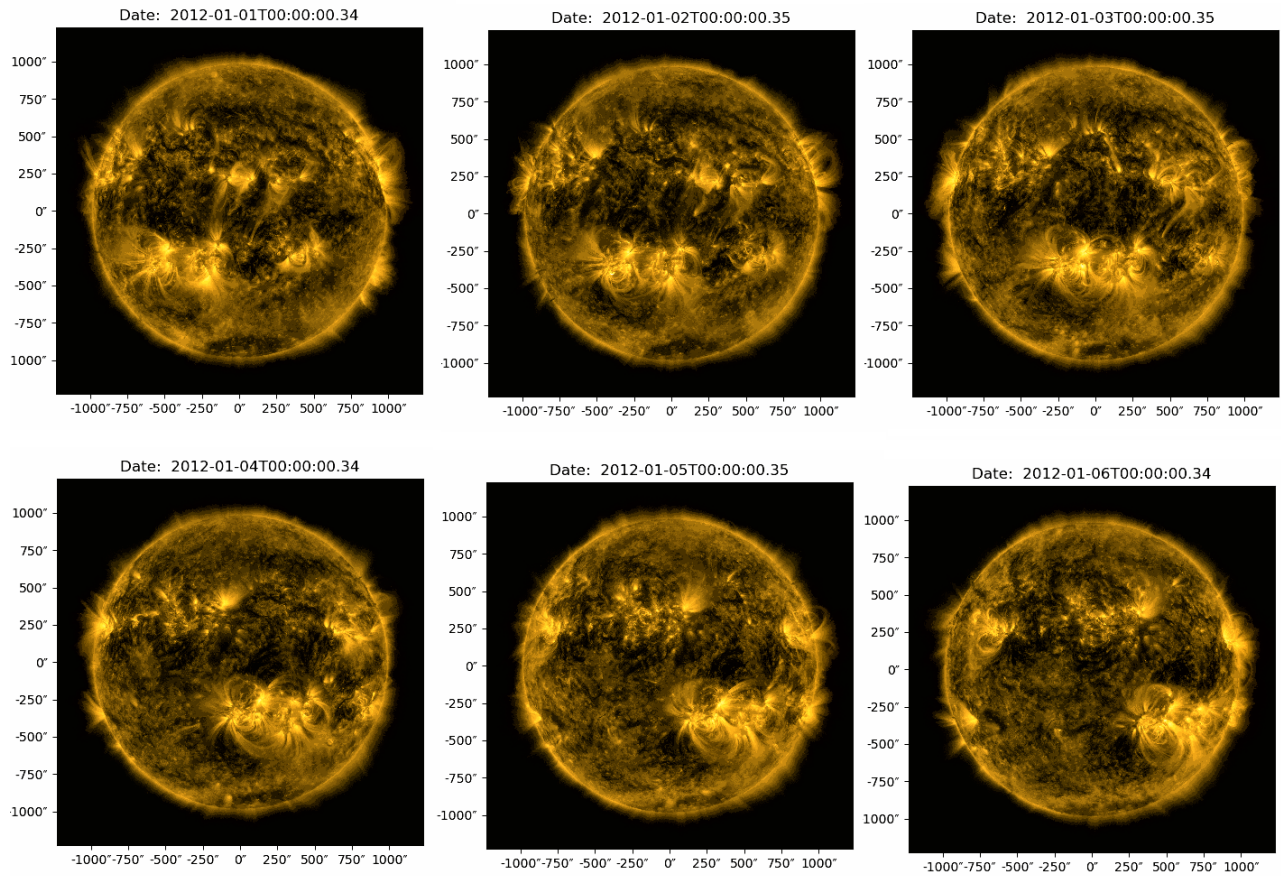
La aplicación ofrece una serie de opciones de modificación de ajustes sobre el *frame* principal cuya función es mostrar las distintas imágenes dentro de la vista. Las opciones de ajustes sobre el mismo son muy diversas, permitiendo hacer uso desde variaciones sobre la escala de los ejes cómo la aplicación de una amplia gama de pseudocolores. Si hacemos hincapié por ejemplo en el abanico de pseudocolores, se puede hacer uso de más de 100 tipos de ellos, pudiendo realizar el usuario dichos cambios según convenga para visualizar de una mejor manera u otra ciertas estructuras sobre la imagen. Un ejemplo de ello sobre la imagen anterior es el que puede encontrarse en la Figura 6.11.



**Figura 6.11:** *Aplicación de diferentes pseudocolores sobre la imagen.*

### 6.3.5. Secuencia de imágenes

Siempre que se quiera cargar una secuencia de imágenes, dicha secuencia se mostrará sobre el panel principal. En ciertas ocasiones, el usuario tendrá la intención de poder almacenar dicha secuencia sobre un formato de vídeo como puede ser un formato GIF (Graphics Interchange Format). Aunque no se pueda mostrar sobre la memoria dicho formato, se simulará una secuencia de la misma dando como resultado el encontrado en la Figura 6.12.



**Figura 6.12:** *Representación de una secuencia de imágenes.*

En ella se representa una secuencia de imágenes del Sol tomadas a 171 Ångstrom entre los días 1 al 6 de enero de 2012.



# Capítulo 7

## Conclusiones y trabajo futuro

### 7.1. Conclusiones

En este trabajo se ha logrado entender la base matemática y el conocimiento necesario para poder diseñar un sistema que sea capaz de realizar un mapa sinóptico mediante una serie de imágenes captadas por una sonda espacial o por cualquier otro sistema desde la Tierra, haciendo uso de distintos instrumentos y técnicas pasabanda.

Se ha realizado una investigación sobre los sistemas de coordenadas heliográficas y en sus posibles proyecciones para lograr un nuevo sistema de coordenadas aplicable directamente sobre una imagen solar. Este logro ha permitido establecer y trazar a su vez el ritmo de composición de las distintas imágenes para lograr dicho mapa.

Para la validación del sistema, se han tomado distintas formas de esclarecer el resultado final teniendo en cuenta el criterio de distintos profesionales en el área. Para ello se registró en que zonas se encuentra el máximo error de cada imagen y en qué manera afecta al mapa completo. Mediante comparaciones en función del tipo de imagen y la banda del espectro en la que fue tomada, se llegó a la conclusión que el resultado obtenido es de buena calidad.

Como resultado final, se ha logrado diseñar una aplicación que permita realizar dichos mapas entre otras muchas funcionalidad, dándole al usuario una herramienta simple y cómoda con la que poder componer y editar un conjunto de imágenes.

## 7.2. Trabajo futuro

Debido a que las imágenes son en su mayoría de alta calidad y de bastante tamaño, se hacen uso técnicas de escalado y *casting* para reducir la carga en la aplicación, aún haciendo uso de compiladores *JIT*. Sería interesante en un futuro poder añadir un sistema de configuración en función de los requisitos del sistema, para ver en qué grado se deberá de aplicar esta optimización sobre el conjunto de imágenes. Por ello, que la primera vía para un trabajo futuro es poder optimizar la aplicación para un uso de imágenes de mayor tamaño y calidad en sistemas con menores requisitos.

Una idea pensada para un futuro próximo a su vez sería poder conectar la aplicación a un sistema *goto* y una cámara astronómica sobre un telescopio, que permita una edición en directo y una forma de observación a tiempo real.

Las imágenes astronómicas en general hacen uso de ficheros *FITS*, y las técnicas usadas en la aplicación también pueden aplicarse a otros sistemas astronómicos, por ello que pueda ser interesante realizar un sistema de seguimiento sobre distintos objetos mediante un telescopio.

Por último, sería interesante poder realizar composiciones multibanda mediante distintas imágenes pasabanda para poder realizar un estudio más completo del Sol, en sus distintas capas que lo componen.

# Bibliografía

- [1] J. M. Vaquero. *The Sun Recorded Through History*. Springer, New York, NY, 2009.
- [2] V. J. Martínez, J. A. Miralles, E. Marco, and D. Galadí-Enríquez. *Astronomía fundamental*. Universitat de València, 2007.
- [3] Universidad de La Laguna. *Galileo y la gestación de la ciencia moderna*. Consejería de Educación, Cultura y Deportes del Gobierno de Canarias, 2000.
- [4] R. Beck, H. Hilbrecht, K. Reinsch, and P. Völker. *Solar Astronomy Handbook*. Willmann Bell Inc., 1995.
- [5] L. Namikawa, E. Castejon, and L. Fonseca. Digital image processing in remote sensing. In *2009 Tutorials of the XXII Brazilian Symposium on Computer Graphics and Image Processing*, pages 59–71, Los Alamitos, CA, USA, oct 2009. IEEE Computer Society.
- [6] A. de la Escalera. *Visión por computador. Fundamentos y métodos*. Prentice Hall, 2001.
- [7] C. Barbieri. *Fundamentals of Astronomy*. New York: Taylor & Francis, 1942.
- [8] J. Ruiz. *El Sol*. Marcombo, 2018.
- [9] P. Galaviz, F. Sánchez-Bajo, and J. M. Vaquero. Determining sunspot positions in the classroom using the carrington method. *European Journal of Physics*, 37(4):045707, jun 2016.
- [10] P. P. J. Pericas. Tres Ábacos para las Mediciones Heliográficas. *Memoires de l’Observatoire de l’Ebre*, 5:1–41, January 1914.

- [11] F. Sánchez-Bajo and J. M. Vaquero. Measuring solar rotation from digital camera images. *European Journal of Physics*, 34(3):527–536, mar 2013.
- [12] J. M. Nogales. *Algunas soluciones cartográficas aplicadas a la física solar*. PhD thesis, Universidad de Extremadura, 2017.
- [13] R. C. Carrington. On the Distribution of the Solar Spots in Latitude since the Beginning of the Year 1854; with a Map. *Monthly Notices of the Royal Astronomical Society*, 19(1):1–3, 11 1858.
- [14] R. C. Carrington, Esq. On a Method of Observing the Positions of Spots on the Sun; with an Example of its Application. *Monthly Notices of the Royal Astronomical Society*, 14(5):153–158, 03 1854.
- [15] E. W. Maunder. Note on the Distribution of Sun-spots in Heliographic Latitude, 1874-1902. , 64:747–761, June 1904.
- [16] A. Hamada, T. Asikainen, and K. Mursula. New homogeneous dataset of solar euvsynoptic maps from soho/eit and sdo/aia. *Solar Physics*, 2019.

# Apéndice A

## Introduction

### A.1. Motivation

During the history of astronomy [1, 2], it is recorded that any celestial data record was stored on paper or stone. The only way to track and study celestial objects until the 17th century was through the observational method, a fact of this is the eastern civilizations, which stored engravings on their observations on the Sun. These civilizations usually used the Sun for the regulation of your calendar. To do this, they observed it when the position of the Sun was close to the horizon as long as appropriate meteorological conditions were met (such as fog or dust storms, among others), causing the atmosphere to act as a solar filter.

With the advance of optics in the late sixteenth and early seventeenth centuries, certain instruments of observation emerged for distant distances, although most were intended for use in war. The invention of the telescope is somewhat fuzzy, it was initially attributed to the German Hans Lippershey, although more recent studies of the 20th century suggest that it was the Catalan Joan Roget. In 1609, after exposing Galileo Galilei, the first refraction telescope oriented to the observation of the stars, to the scientific community of the time, a new era of discoveries arises in astronomical history, with new fields of study and research applied to the Sun.

Throughout the 17th century, different types of telescopes began to emerge, all of which



followed similar patterns, although the arrangement of their lenses varied. After that, in 1611 the sunspot race began. The German Christopher Scheiner invented various techniques for observing the Sun, focusing on methods that allowed it to be viewed in a safe way (by means of filters or projections on a membrane, as if it were a slide).

Scheiner supposed that these spots must be projections of stars on its surface or defects in the eye itself or the lens when observing. Back then it was suggested that every heavenly body was perfect. Instead, Galileo described it as large bumps on the surface of the Sun that had motion. This thought led him to the right thing, allowing him to calculate the first astronomical data of the Sun such as its period and its speed of rotation among others [3].

More complete and precise studies on sunspots will not begin until almost 2 centuries forward with the German Heinrich Schwabe. Schwabe made daily sunspot records from 1826 to 1843. During this time, he discovered a periodicity in the number of sunspots every 11 years which he called the cycle. This fact gave rise to the need to quantify solar activity, not only from the 19th century, but from previous years, making use of notes and notebooks from around 1745 to be able to quantify and explain its finding with some precision.

The Swiss Johann R. Wolf intrigued, began in 1847 a counting of spots and pores, to which as a whole he assigned an easily calculable value, known as Wolf's number. Although there are variations such as the group count (1998), both are usually applied today. Wolf's number allows abstracting the observations of each day into a number, recording only one value per observation.

On September 1, 1859 the largest possible solar storm in history is recorded by Richard C. Carrington (this event is known as the Carrington event). When observing, a large explosion is discovered for the first time next to the exit of a large flare from the Sun, known as CME (Coronal Mass Ejection).

After observing this event, the following day a geomagnetic storm was detected on Earth accompanied by auroras at low latitudes, indicating to 19th century scientists that solar ejections could apparently be related to these phenomena. This phenomenon will be studied

later in 1931 by Sydney Chapman and Vincenzo Ferraro. Carrington conducted extensive research with incredible accuracy, allowing his techniques to continue to be used today. Among them are: the heliographic coordinate system and the differential rotation depending on the latitude of the body. These two topics will be addressed throughout the memory.

One of the greatest advances in the field of solar astronomy was made by the American astronomer George Ellery Hale, who in 1892 invented the spectroheliograph, an instrument that allows observing the Sun in a band of the electromagnetic spectrum. Hale continued in parallel with the study of geomagnetic storms and concluded by reading heliomagnetograms in 1903 that sunspots are associated with magnetic fields. Since then, a wide variety of new studies have emerged, leading to research such as solar wind, cosmic radiation, auroras, X-ray emissions, etc.

From the discovery of the solar cycles to the magnetic activity of the Sun, among many other varied studies, in order to compare results in a more optimal way in all of them, whether by drawing or reading magnetograms, it is recorded on a single map or various diagrams. rotations of the sun allowing to compare its observable elements in different periods of time. These representations are commonly known as solar planispheres or synoptic maps of the Sun and it is the purpose of this project.

## A.2. Objectives

The objective of this work is the development of an application that allows to manage solar images and develop synoptic maps. This application will have various tools that allow you to reach the described task in a simple and powerful way.

To achieve the final objective, the following specific objectives will be addressed in the report, which are listed below:

- File manipulation *FITS* (Flexible Image Transport System).
- Analysis and understanding of the different positioning systems on the Sun.

- Application of different coordinate systems to place a structure on an image.
- Development of a system capable of creating synoptic maps from a sequence of images.
- Validation of synoptic maps obtained from similar images by experts.
- Development of an application capable of composing synoptic maps and having a series of simple editing tools.

### A.3. Work plan

The work will be structured by different main activities dictated by the objectives to be met, which will be complemented throughout the development of the project. These activities will be divided into four main groups, among them:

- *Initial phase*: the general structure of the project will be described.
- *Research phase*: the necessary information will be obtained and put into practice for the proper development of the memory and the programming project.
- *Application development phase*: the different components of the application will be developed.
- *Memory development phase*: the different parts that will compose the memory will be written.

### A.4. Organization of this memory

The report will be structured in the following main chapters, each collecting the following competencies:

- **Introduction**: a brief historical analysis of solar physics will be carried out, the objectives of the project will be shown and the different competences provided by each chapter will be analyzed.

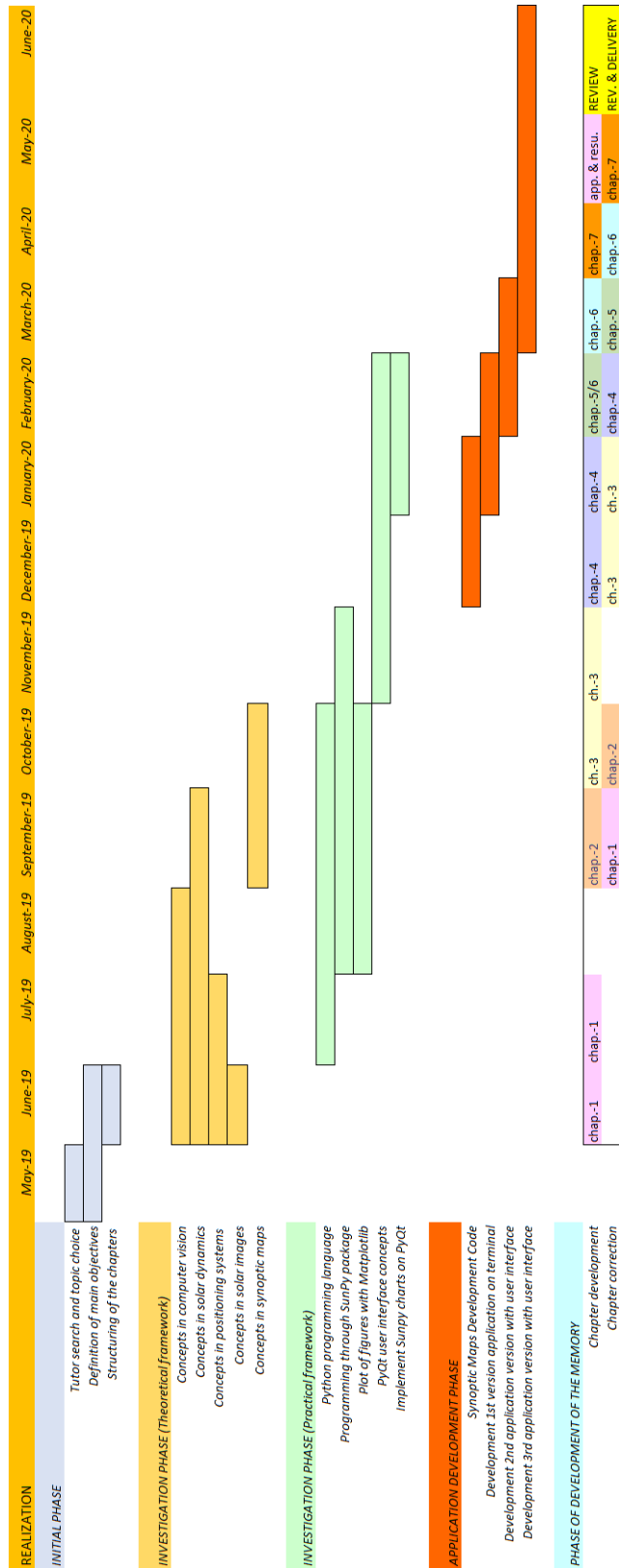


Figura A.1: *Work plan.*

- **Analysis of solar images:** it will be in the second chapter where astronomical images and the *FITS* format used for storage will be discussed. The sensors used to obtain the images will be explained.
- **Coordinate systems and image orientation:** in the third chapter mention will be made of geocentric and heliographic coordinate systems, the movements that affect the Sun and the different ways of projecting the different coordinate systems. In turn, it will be shown how to position a point on a solar image and how to orient it properly.
- **Synoptic Maps:** Chapter 4 will refer to the projection systems in cylindrical coordinates and the use of systems based on *squircle* to make such projections on solar images. The method of composition and validation of these maps will be taught.
- **Application development:** the necessary elements for the application's own structure will be discussed throughout Chapter 5.
- **Functionality of the application:** the main functionalities and tools available to the application will be shown. All of this will be described in Chapter 6.
- **Conclusions and future work:** they will be described in the last chapter. The different conclusions and facts carried out throughout the Final Degree Project and their possible implementations for future works will be shown.



# Apéndice B

## Conclusions and future work

### B.1. Conclusions

In this work, it has been possible to understand the mathematical basis and the knowledge necessary to design a system that is capable of making a synoptic map using a series of images captured by a space probe or by any other system from Earth, using different band-pass instruments and techniques.

An investigation has been carried out on heliographic coordinate systems and their possible projections to achieve a new coordinate system applicable directly to a solar image. This achievement has made it possible to establish and trace the composition rate of the different images in order to achieve said map.

For the validation of the system, different ways have been taken to clarify the final result taking into account the criteria of different professionals in the area. For this, it was recorded in which areas the maximum error of each image is found and how it affects the complete map. Through comparisons based on the type of image and the band of the spectrum in which it was taken, it was concluded that the result obtained is of good quality.

As a final result, it has been possible to design an application that allows making such maps, among many other functions, giving the user a simple and comfortable tool with which to compose and edit a set of images.

## B.2. Lines of future work

Because the images are mostly high quality and quite large, scaling and *casting* techniques are used to reduce the load on the application, even using *JIT* compilers. It would be interesting in the future to be able to add a configuration system depending on the system requirements, to see to what extent this optimization should be applied to the set of images. Therefore, the first way for future work is to be able to optimize the application for the use of images of a larger size and quality in systems with lower requirements.

An idea thought for the near future in turn would be to be able to connect the application to a system *goto* and an astronomical camera on a telescope, which allows live editing and a form of real-time observation.

Astronomical images generally make use of *FITS* files, and the techniques used in the application can also be applied to other astronomical systems, so it may be interesting to carry out a tracking system on different objects using a telescope.

Finally, it would be interesting to be able to make multi-band compositions using different band-pass images to be able to carry out a more complete study of the Sun, in its different layers that compose it.





# Apéndice C

## Estudios reales haciendo uso de la aplicación

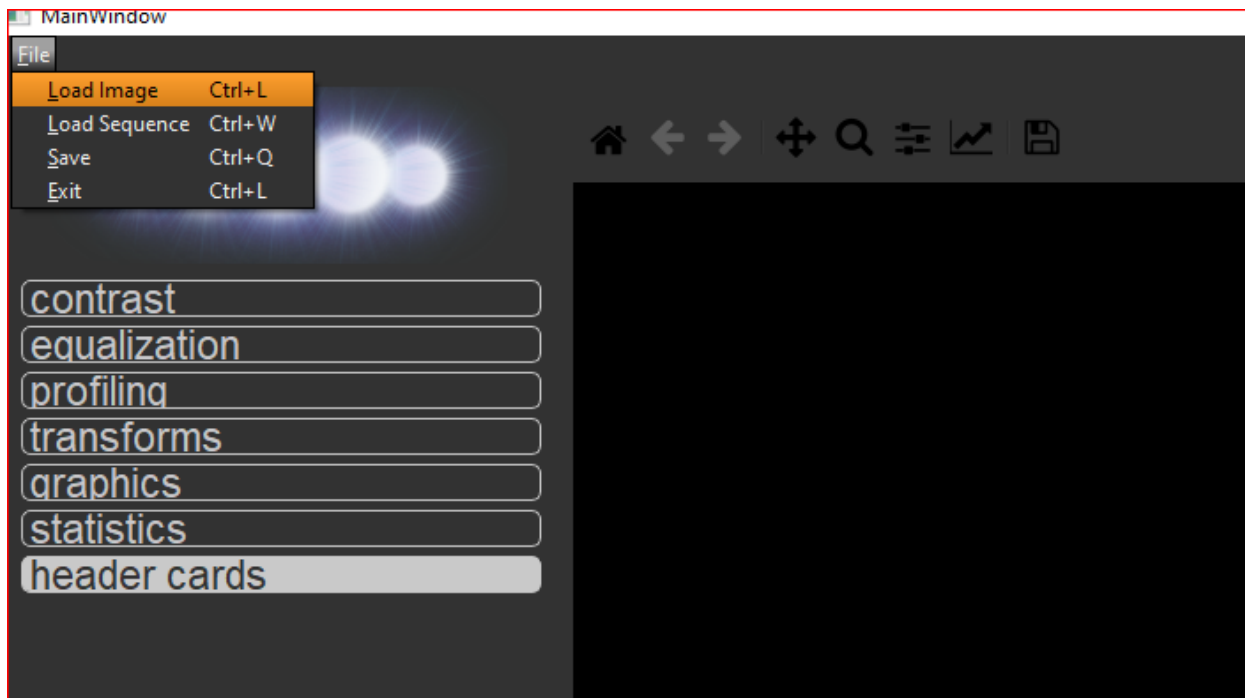
Se realizará una serie de ejemplos reales sobre la aplicación creada. Entre ellos se realizarán: un procesamiento de nitidez, un perfilado de las distintas estructuras encontradas sobre la imagen y el procesado de un mapa sinóptico.

### C.1. Mejora de nitidez sobre imagen SDO/AIA

Para este estudio se tomará una imagen procedente del instrumento SDO/AIA sobre una longitud de onda de 193 Ångstrom, con la que se podrá observar las distintas estructuras posibles con este instrumento sobre la corona solar (para las eyecciones de masa coronales o EMC se usan otros instrumentos como el STEREO con el que se puede abarcar un mayor área de visualización).

La longitud de onda de 193 Ångstrom coincide en parte con el espectro de emisión del  $Fe_{12}$  y con el espectro de emisión del  $Fe_{24}$ . Para que un átomo de Hierro pueda ionizarse hasta el ion  $Fe_{12}$ , las temperaturas que tienen que alcanzarse deben de ser al menos de  $10^6 K$  y para que se pueda ionizar hasta el  $Fe_{24}$ , las temperaturas subirán hasta los  $24 \cdot 10^6 K$ . En la banda del espectro en la que emite el  $Fe_{12}$  se podrá observar sobre la imagen las zonas de la corona solar, que es la capa más alejada del centro del Sol, y las del  $Fe_{24}$  corresponderán con las erupciones solares.

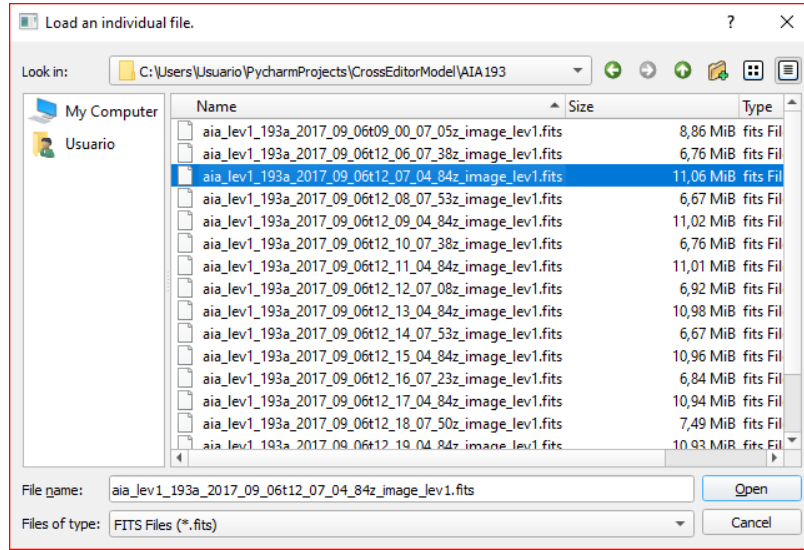
El primer paso para poder visualizar la imagen sobre la aplicación será abrir el menú superior izquierdo *File* y pulsar *Load Image* tal y como se muestra en la Figura C.1.



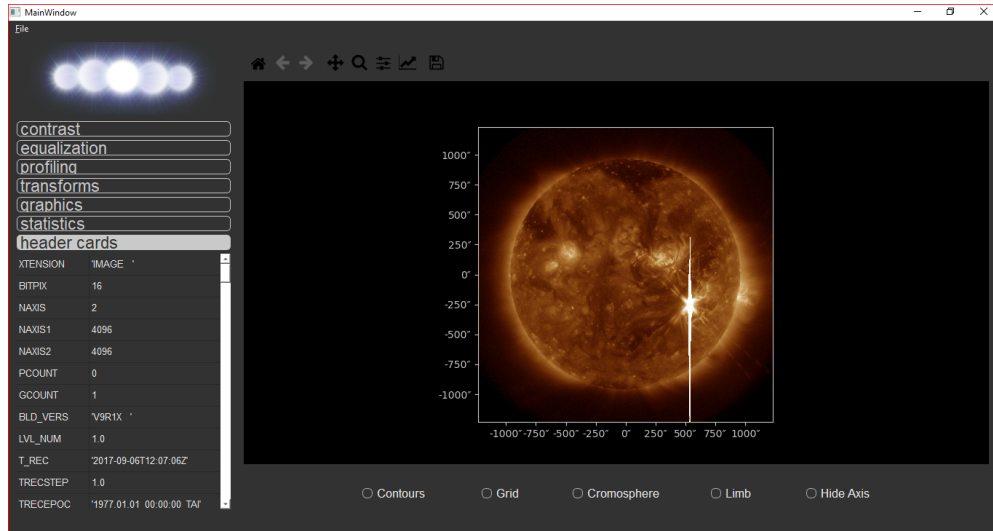
**Figura C.1:** *Menú de carga.*

Una vez pulsado, se abrirá un diálogo en el que poder elegir el fichero que se quiera cargar (ver Figura C.2). En este caso se escogerá una imagen tomada el 6 de septiembre de 2017 visualizándose en el panel derecho de la aplicación (ver Figura C.3). La imagen FITS que será cargada viene por convenio con un pseudocolor específico del instrumento que permite identificarla de las demás, aunque la aplicación permita posteriormente hacer uso de otra distinta en cualquier momento. Para el caso de esta imagen, el pseudocolor albergará un gradiente de tonalidades sobre blancos y marrones.

Esta imagen en especial impresiona debido a que contiene una de las llamaradas o fulguraciones registradas más potentes de los últimos 20 años. Estas llamaradas son clasificadas haciendo uso de una letra y un número. La letra indica en que grupo se incorpora en función de la energía emitida por metro cuadrado y el número, que potencia ha tenido dentro de su grupo. Este fenómeno se ha clasificado como grupo X9.3, indicando que es 9.3 veces



**Figura C.2:** Elección de fichero FITS.

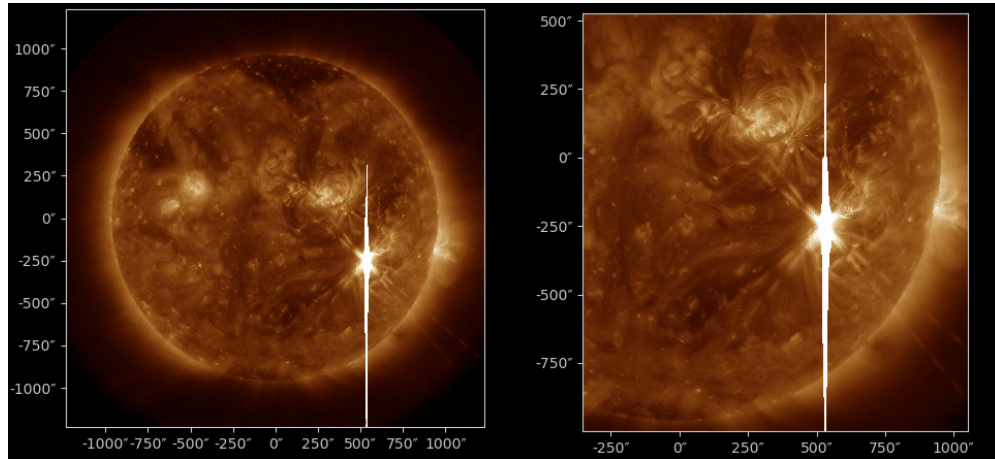


**Figura C.3:** Muestra en el frame el fichero FITS escogido.

más fuerte que una del grupo X1. Usualmente estas grandes explosiones emiten una gran cantidad de materia al exterior del Sol y que gran parte de dicha materia se mueve haciendo uso de los vientos solares que se originan en los huecos o agujeros de la corona solar. Estas grandes eyecciones de masa coronales son comúnmente conocidas como EMC.

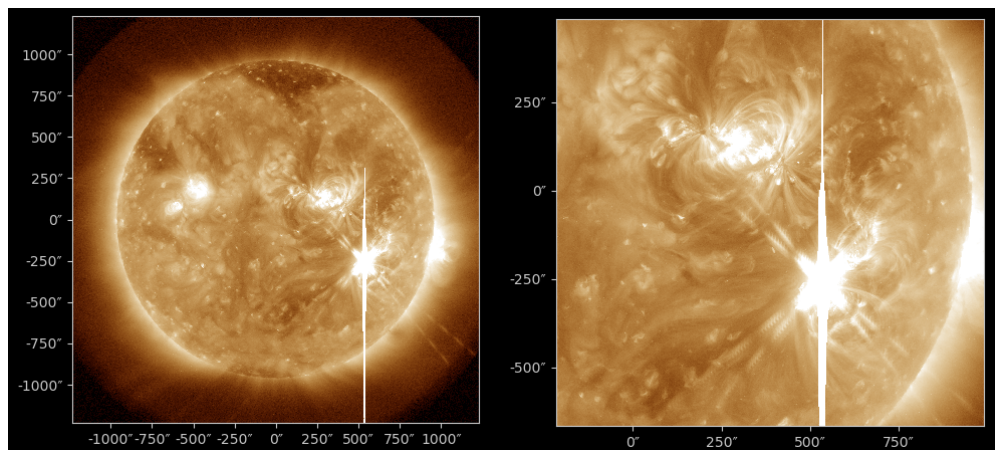
Para obtener un mejor detalle sobre la imagen original (ver Figura C.4), se tratará en los siguientes pasos de sacarle una mayor nitidez que permita observar todas sus estructuras

visibles de un mejor modo.



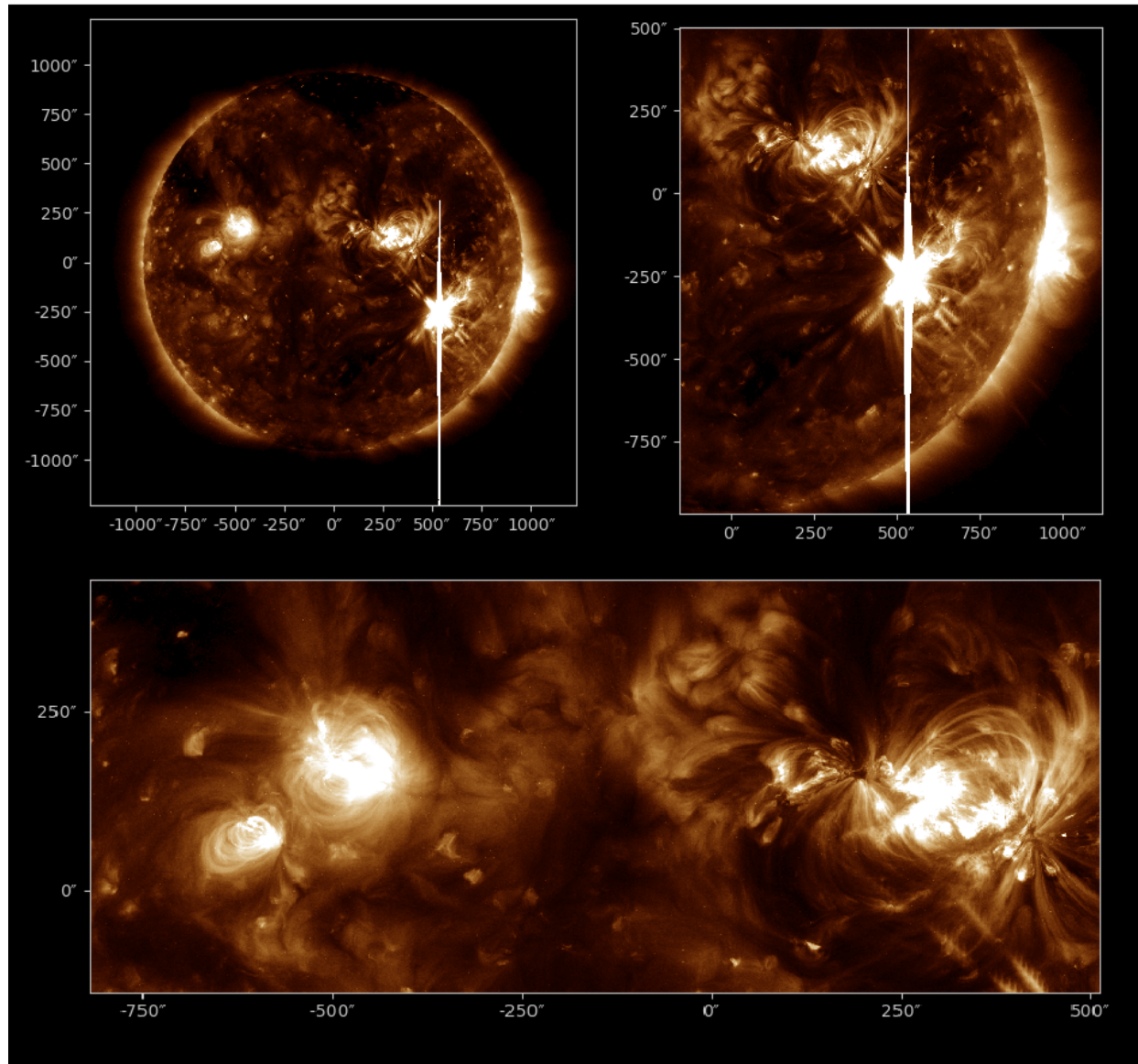
**Figura C.4:** *Imagen base sobre la que se realizará el procesamiento de mejora de nitidez.*

Para ello se comenzará aplicando un filtro selectivo (el cual se podrá encontrar dentro de la aplicación en la parte de *equalization* -> *Contrast stretching*). Este filtro permitirá subir la intensidad asociada a cada píxel que supere un valor o umbral, haciendo de cota de intensidad. Dicho umbral será un porcentaje de la intensidad máxima que pueda adoptar un píxel en función del número de bits por el que esté definido. Si se aplica este método a la imagen cargada cuyos píxeles son representados con 16 bits sin signo (uint16) y cogiendo un umbral del 99 %, la imagen transformada puede observarse en la Figura C.5.



**Figura C.5:** *Mejora de brillo selectivo sobre la imagen base.*

Como resultado, se ha intensificado aquellas zonas en la que existía una variación de intensidad levemente contrastada. Aunque la imagen sea bastante mas luminosa que antes, tras aumentar el rango del contrastado, si se aplica posteriormente una corrección gamma, el contraste será mayor consiguiendo ese efecto de aumento de nitidez en comparación a la imagen base. Para ello, se escogerá un parámetro gamma igual a 2.5, dando como resultado final del procesado el encontrado en la Figura C.6.

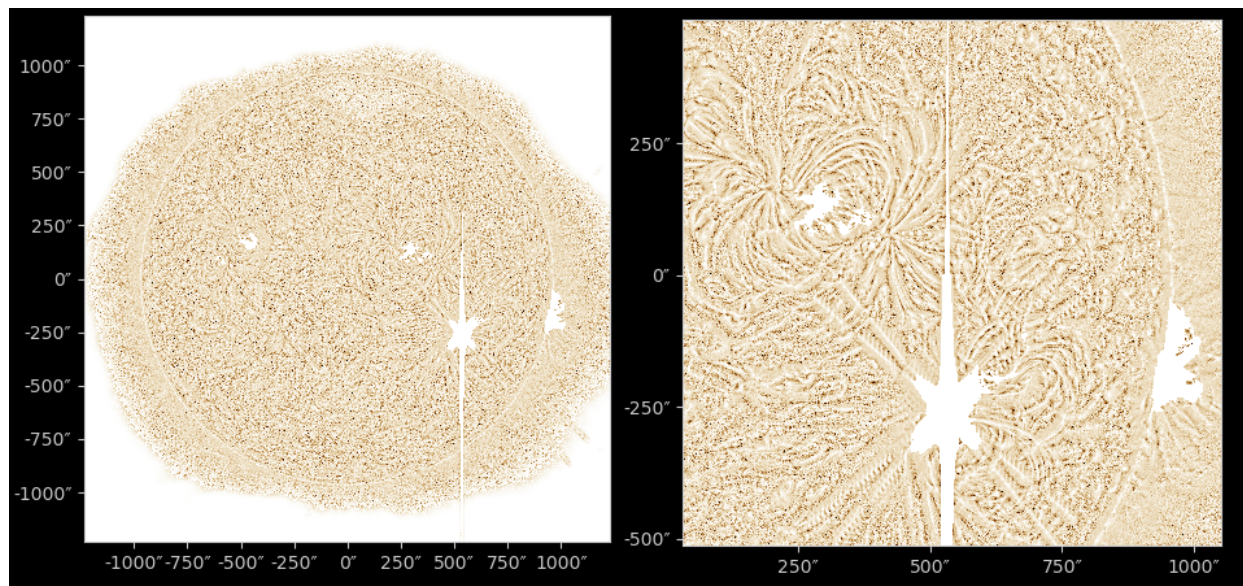


**Figura C.6:** *Diferentes secciones del resultado tras el procesado de nitidez.*



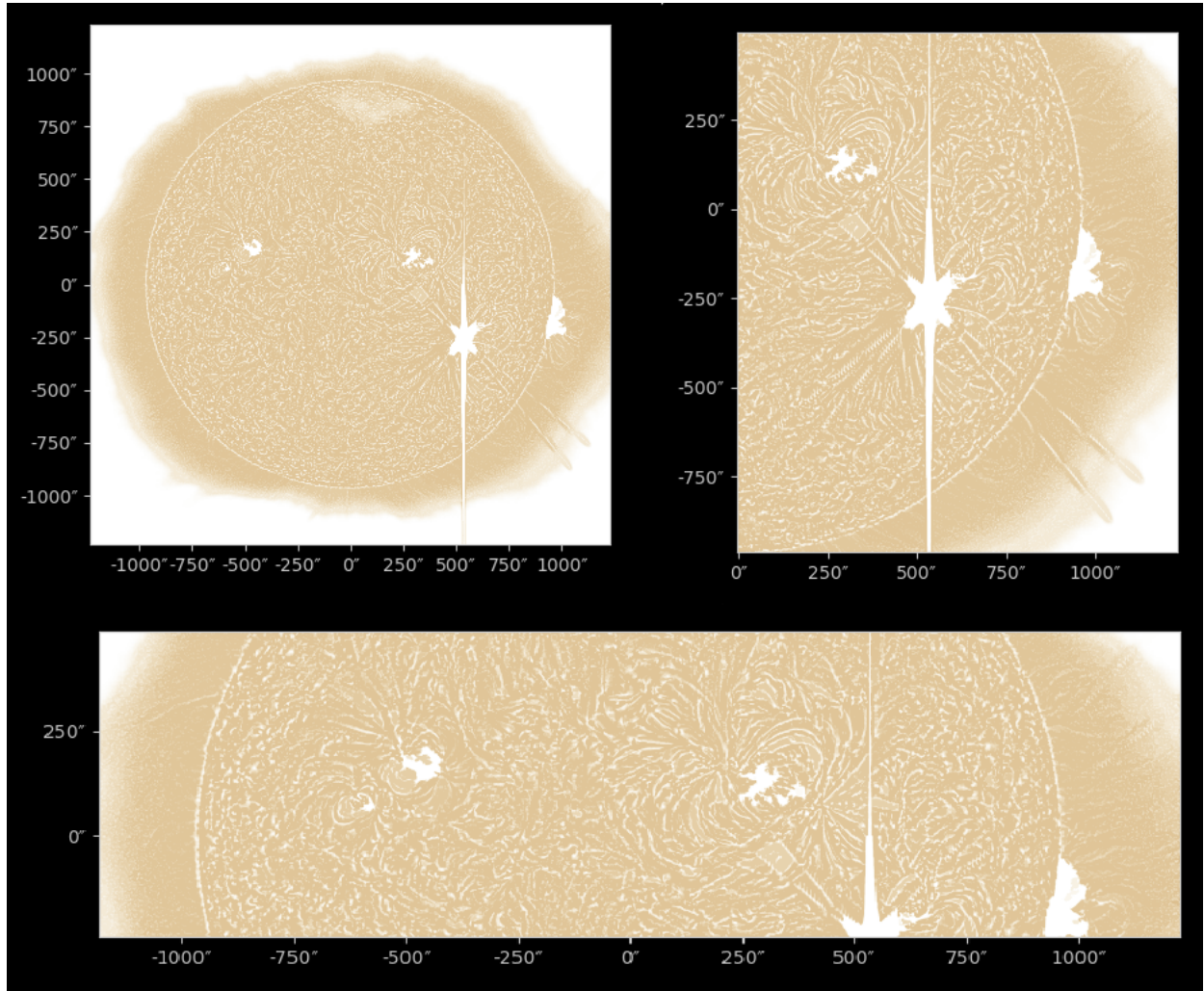
## C.2. Perfilado de estructuras

Una vez se tiene las estructuras resaltadas, el siguiente paso será un procesado de perfilado. Este procesado permitirá visualizar los distintos arcos de plasma y separarlos de la imagen principal para una completa visualización de sus estructuras. El primer filtro que se aplicará será una ecualización local de la imagen, usando un área de tamaño parámetro, dado por el usuario, como máscara local. Dicho filtro es encontrado en la interfaz pulsando en *equalization* -> *Rank equalization*. Este filtro local actuará sobre los elementos luminosos que abarcan los arcos y los destellos encontrados en la imagen y dejará con poca intensidad los elementos restantes. Al aplicarlo con un parámetro 15, queda como resultado el encontrado en la Figura C.7.



**Figura C.7:** Aplicación de una ecualización local de parámetro 15.

El resultado a primera vista es similar a un emboss, aunque ya empieza a detectarse todas las estructuras. Por ello que para resaltarlas aun más, se realizará una binarización. La imagen resultante se puede apreciar en la Figura C.8.



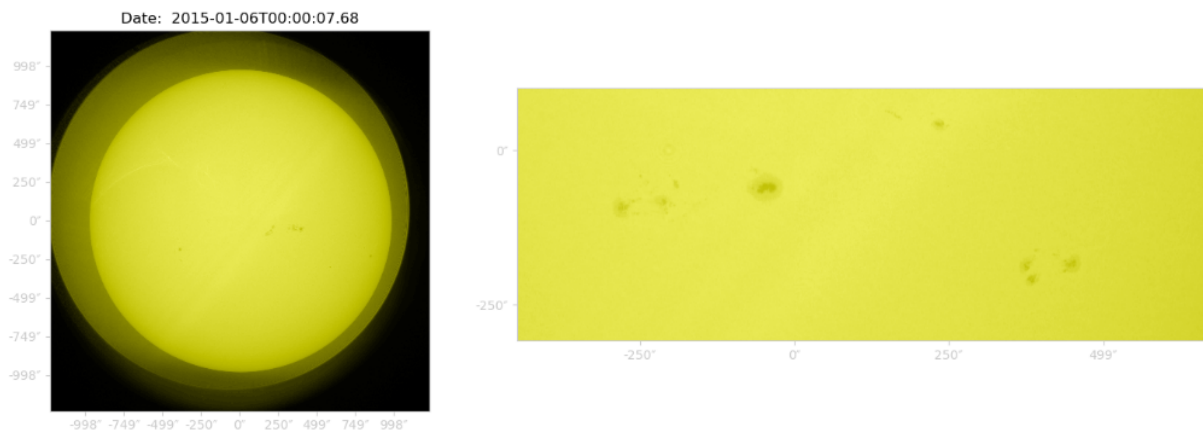
**Figura C.8:** *Resultado final del perfilado de las distintas estructuras encontradas en la imagen tratada.*

### C.3. Procesamiento de un mapa sinóptico

Durante toda la memoria, se ha ido tratando de como crear los mapas sinópticos paso por paso, llegando a un resultado funcional. Para este ejemplo de uso real se harán uso de imágenes tomadas entre el día 1 al día 31 de enero del año 2015, todas ellas tomadas por el instrumento SDO/AIA a 4500 Ångstrom, cayendo dicha longitud de onda dentro del espectro visible. Las imágenes observadas aparecerán de color amarillo y en ellas se podrá observar la fotosfera (superficie luminosa cuya temperatura ronda entre los 4000 a 6000 K)

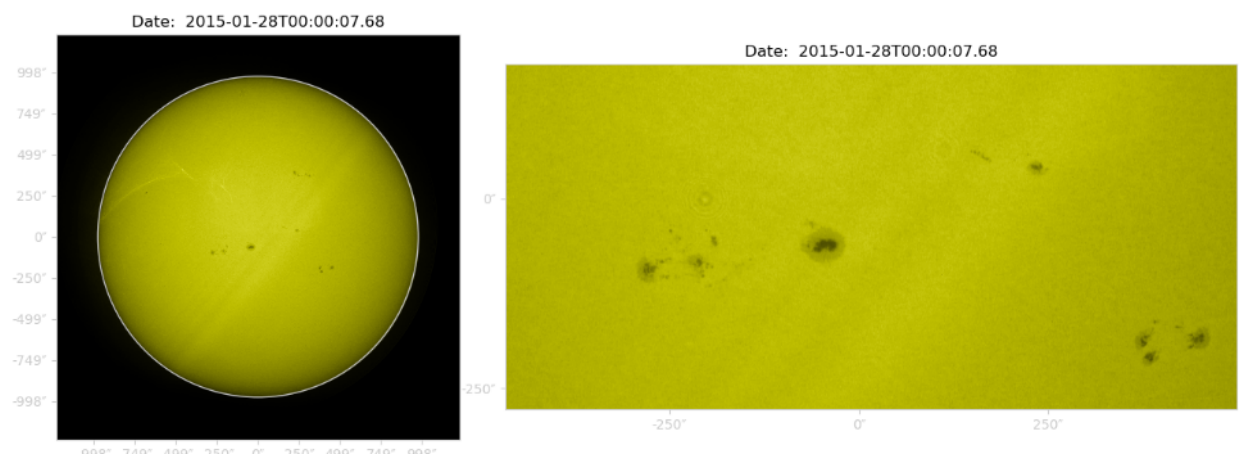


junto a sus manchas solares. Un ejemplo de este tipo de imágenes es el observable en la Figura C.9.



**Figura C.9:** *Imagen procedente del instrumento SDO/AIA 4500 Ångstrom tomada el día 06/01/2015.*

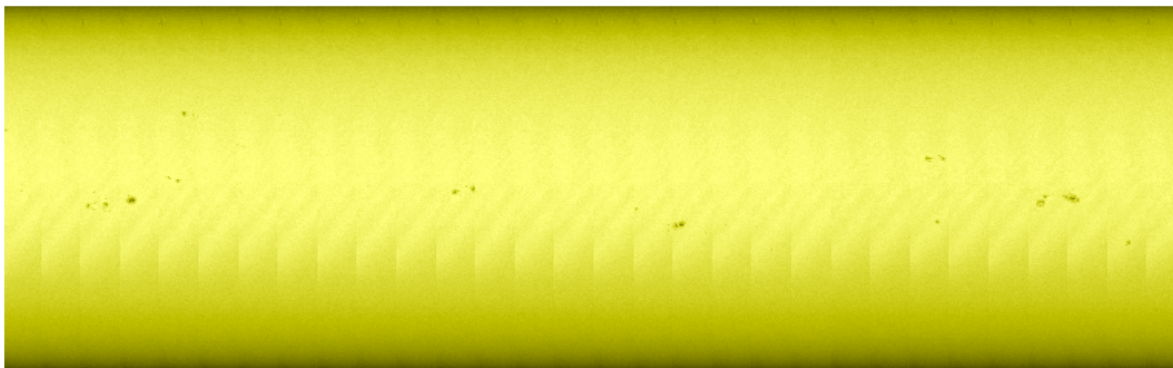
Debido a la luminosidad de la imagen, las manchas no suelen apreciarse bien a simple vista, por ello que haya que aplicarle en la mayoría de los casos una corrección de luminosidad o un cambio de pseudocolor para mejorar su visualización. El resultado puede encontrarse en la Figura C.10.



**Figura C.10:** *Aplicación de un ajuste gamma para reducir el exceso de luminosidad.*

Para realizarlos, el primer paso es cargar una secuencia de imágenes, similar al de cargar

una única imagen. Para ello, se seleccionará *Menú File ->Load Sequence*. Una vez pulsado el botón, saldrá el buscador para que le introduzcamos una ruta de un directorio. Tras haber cargado las imágenes como una secuencia, el último paso es pulsar *transforms ->Synoptic Map*. dando como resultado el encontrado en la Figura C.11.



**Figura C.11:** *Aplicación de un ajuste gamma para reducir el exceso de luminosidad.*